Maggio/Giugno 1988, Anno III n. 3
Mensile – Lire 7.000

La rivista per utenti di C-64/128 ed Amiga

COMMODORE GAZETTE

Dal nostro inviato a Washington:

L'AMIGA DEL FUTURO ALL'ANNUALE CONVEGNO DEI PROGRAMMATORI

Prove software:

GEOPROGRAMMER

Prove hardware:

I NUOVI PC MS-DOS DELLA COMMODORE

Telematica:

LE RETI A PACCHETTO

Sped. in abb. post. Gr. III/70

Articoli

MAGGIO/GIUGNO 1988
Anno III, N.3

Rubriche

La rivista per utenti di C-64/128 ed Amiga

COMMODORE GAZETTE

Una pubblicazione

Direttore Responsabile
Massimiliano M. Lisa

Redazione
Luca Giachino (capo servizio redazione tecnica)
Mauro Gaffo (redattore)
Nicolò Fontana Rava (servizi speciali)
Collaborazione Editoriale
Alfredo Macchi, Marco Menichelli, Sergio Fiorentini, Simone Majocchi, Franco Toldi
Corrispondenti USA
William S. Freilich (sezione sviluppo)
Daniela D. Freilich (coordinamento generale)
Collaborazione Editoriale USA
Louis R. Wallace, Ervin Bobo

Inviato speciale USA
Matthew Leeds

Segretaria di Redazione
Susan Visentin

Impaginazione e Grafica
Antonio Gaviraghi (capo servizio)
Andrea De Michelis
Fotografia
A.&G.,
Patricia Leeds
Disegni
Marco Piazza, Giuseppe Festino, Lara Parravicini
Word Processing
Loretta Giovacchini, Fabia Gorini

Direzione, Redazione, Amministrazione
IHT Gruppo Editoriale S.r.l.
Via Monte Napoleone, 9 - 20121 Milano

Fotocomposizione
IHT Gruppo Editoriale S.r.l. - Reparto grafica
Fotolito
Litoscan Bencetti s.r.l.
Via Gallarate, 116 - Milano
Stampa
Rotolito Lombarda S.p.A.
Via Brescia, 53/55 - Cernusco sul Naviglio

Distribuzione per l'Italia
Messaggerie Periodici S.p.A.
V.le Famagosta, 75 - 20142 Milano
Tel. 02/8467545

Distribuzione per l'estero
A.I.E. S.p.A.
Via Gadames, 89 - 20151 Milano
Tel. 02/3012200

Pubblicità
IHT Gruppo Editoriale S.r.l.
Via Monte Napoleone, 9 - 20121 Milano
Ufficio Pubblicitario IHT
Italia ed Estero 02/794181-799492-792612-794122
Fax 784021 - Telex 334261 IHT I

Servizio Abbonamenti

IHT Gruppo Editoriale
Servizio Abbonati
Via Monte Napoleone, 9 - 20121 Milano

SEGRETERIA ABBONAMENTI
Linea per registrazione abbonamenti
02/794181-799492-792612-794122

Commodore Gazette
Costo abbonamenti

Italia
12 numeri L. 84.000
24 numeri L. 168.000
36 numeri L. 252.000
Estero:
Europa L. 120.000 (10 numeri)
Americhe, Asia... L. 180.000 (10 numeri)

I versamenti devono essere indirizzati a
IHT Gruppo Editoriale S.r.l.
Via Monte Napoleone, 9 - 20121 Milano
mediante emissione di assegno bancario o vaglia postale

Arretrati
Ogni numero arretrato: L. 14.000 (sped. compresa)

Autorizzazione alla Pubblicazione
Tribunale di Milano n. 623 del 21/12/85
Periodico Mensile - Sped. in abb. post. gr. III/70
ISSN: 0394-6991

La IHT Gruppo Editoriale è iscritta nel Registro Nazionale della Stampa al n. 2148 vol. 22 foglio 377 in data 5/6/1987

Commodore Gazette è una pubblicazione
IHT Gruppo Editoriale

Associato alla U.S.P.I. (Unione Stampa Periodica Italiana)

NOTE EDITORIALI

A partire dal mese prossimo la Divisione libri della IHT Gruppo Editoriale, la casa editrice di *Commodore Gazette*, pubblicherà numerosi nuovi titoli. Ai primi di luglio usciranno in libreria due nuovi testi della collana informatica che si aggiungeranno ai precedenti *L'Amiga*, *Il Manuale dell'AmigaDOS* e *Flight Simulator Co-Pilot*. Sono la *Guida ufficiale alla programmazione di GEOS*, un libro per gli appassionati di GEOS e per chi desidera imparare a programmare in questo nuovo ambiente operativo, e *Volare con Flight Simulator*, un testo dedicato alla simulazione del volo su Amiga, Atari ST e Apple Macintosh.

Ai primi di ottobre uscirà *Programmare l'Amiga Vol. II*, che sarà seguito da *Programmare l'Amiga Vol. I* nel marzo '89. In aprile usciranno due volumi dedicati al nuovo sistema operativo della Microsoft per gli IBM PS/2: *Il manuale dell'OS/2* e *Programmare in OS/2*.

Verso la fine del 1989 saranno pubblicati quattro libri sulla programmazione dell'Amiga ideati e realizzati dalla stessa Commodore-Amiga, che, integrandosi con i volumi precedenti, costituiranno la biblioteca indispensabile per chi desidera sviluppare software sull'Amiga.

La IHT ha anche deciso di estendere la sua attività editoriale al vasto campo della scienza e della tecnologia. La prima iniziativa in questo senso è la collana TEMPUS, che sarà dedicata alla storia, all'impatto sociale e al futuro della scienza e dell'alta tecnologia. Sono in programma volumi che toccando gli argomenti più diversi costituiranno una biblioteca d'alto livello per quei lettori che vogliono tenersi aggiornati con la rivoluzione informatica, ma anche con i progressi del mondo della fisica, della matematica, dell'economia. Verranno presentati testi che uniscono facilità di lettura e profondità di prospettiva, che esplorano la scienza e la tecnica dei giorni nostri con lo sguardo sempre rivolto verso il domani.

FOTO PATRICIA LEEDS

Le prime uscite della collana sono previste a partire dal mese di novembre, e nell'ordine sono: *Inventori del nostro tempo*, interviste con 16 famosi inventori americani; *Computer in guerra: funzioneranno?*, sui rischi e le potenzialità delle nuove tecnologie militari; *La macchina e la mente*, alla scoperta della nuova scienza dell'intelligenza artificiale; *I creatori del domani*, sul mondo nuovo delle macchine organiche; *Il segreto del successo*, sulle sfide, le opportunità e i pericoli degli investimenti nell'economia di oggi.

SoftMail News

® SoftMail è un marchio registrato da Lago

Grazie ai numerosissimi clienti affezionati e a coloro che ogni giorno si aggiungono al nostro mailing-list, SoftMail è diventata la maggiore organizzazione di vendita per corrispondenza di software ed accessori per computers in Italia.
Ecco alcune informazioni utili per chi vuole usufruire del servizio SoftMail: è possibile effettuare ordini telefonici SOLO se è già stata effettuata una spedizione a proprio nome ed è stata regolarmente ritirata. Dal secondo in poi accettiamo anche ordini telefonici. Se avete effettuato un ordine e vi interessa sapere se e quando vi è stato spedito, il nostro servizio on-line vi darà ogni informazione ESCLUSIVAMENTE di POMERIGGIO dalle 14:30 alle 16:30. Chi invece desidera ricevere informazioni più complete, può chiamare quando vuole.
Le spese di spedizione sono gratuite se l'ordine supera Lit. 50.000; per ordini inferiori a tale importo il concorso spese è di Lit. 5.000. Nota bene: i programmi che alla ricezione dell'ordine non sono ancora in commercio verranno spediti non appena disponibili con un addebito per le spese di spedizione di Lit. 3.500.

ACCESSORI

Copritastiera A500	25.000
Drive esterno Amiga	299.000
Final Cartridge III	112.100
Joy. IconTroller	35.000
Joy. SpeedKing	29.000
Joy. Tac 5	39.000
MouseMat-tappetino	22.500
MouseMouse-coprimouse	20.000
Portadischi 3" (30)	34.000
Portadischi 5" (40)	37.000

AMIGA

Aargh!	39.000
Acquisition 1.3	450.000
Alt. reality: City	69.000
Arkanoid (PAL)	59.000
Armageddon man	49.000
Beyond Zork	49.000
Black lamp	39.000
Blackshadow	49.000
Bubble Bobble	29.000
Butcher	59.000
C64 Emulator	telef.
Capone	59.000
Corruption	telef.
Crack	35.000
Crystal hammer	29.000
Destroyer	29.000
DigiView 3.0	telef.
Director	99.000
Ebonstar	45.000
Enlightenment	39.000
Faery Tale Hints	10.000
Flight Simulator II	99.000
Interact.cable A500	20.000
Scenery disks	telef.
Scenery notebook	15.000
Flinstones	39.000
Forms in flight	125.000
Galileo 2.0	99.000
Gettisburg..point	99.000
Grabbit	49.000
Jet	99.000
Mission elevator	29.000
Obliterator	45.000
Phantasie III	49.000
Photon paint	155.000
Ports of Call	69.000
Prof. Page 1.1	499.000
Publisher Plus	125.000
Rockford	39.000
Romantic encounters	49.000
Sentinel	39.000
Shakespeare	250.000
Space port	29.000
Starfleet	59.000
Strike force harrier	49.000
Tetris	39.000
The Works! (integr)	249.000
Thexder	59.000
Univ. Military Sim.	telef.
Video Wizard	49.000
VizaWrite 1.09	159.000
Wizball	49.000
Write & File	125.000

COMMODORE 128 (80 COLONNE)

Basic 8.0	75.000
Basic compiler	99.000
"C" Language	99.000
Cobol	79.000
Desk Pack Plus/GEOS	60.000
Font Pack Plus/GEOS	60.000
Fontmaster w/spell	99.000
Stratton	29.000
Thai boxing	15.000
VizaStar 128	135.000
VizaStar man. ital.	15.000
VizaWrite Classic	125.000
VizaWrite man. ital.	10.000

COMMODORE 64/128 (CASSETTA)

Apollo 18	22.000
Basket master	12.000
Chessmaster 2000	22.000
Flight Simulator II	39.000
Gothic	10.000
Match day II	12.000
Morpheus	29.000
Pandora	10.000
Platoon	10.000
Power at sea	22.000
Predator	10.000
Rimrunner	10.000
Samurai warrior	10.000
Skate or die	22.000
Test drive	22.000
The Bard's Tale I	telef.
Troll	10.000
Vixen	22.000
Wolfman	22.000

COMMODORE 64/128 (DISCO)

Apollo 18	29.000
ATF Adv. Tact. Figh.	25.000
DeckerBASIC/GEOS	79.000
Deja Vu	29.000
Earth orbit station	35.000
Flight Simulator II	99.000
FS accessori vedi AMIGA	
Font Pack 1	45.000
Frankenstein	29.000
Gee Bee Air Rally	29.000
Gunship	35.000
Ikari warriors	19.900
Intrigue	25.000
Knight & Magic	telef.
Jack the ripper	29.000
Jet 2.0	89.000
Jinxter	35.000
Legacy of Ancients	35.000
Maniac mansion	49.000
Morpheus	35.000
Patton vs Rommel	35.000
Pers.Newsletter/GEOS	79.000
Project: Stealth F.	35.000
Rimrunner	25.000
SS: Soccer (USA)	telef.
Stealth Fighter	75.000
Strikefleet	29.000
Tetris	25.000
The Bard's Tale I	29.000
The Bard's Tale II	29.000
The Bard's Tale III	telef.
The train	29.000
Troll	25.000
Vixen	29.000
WWF Micro. Wrestling	39.000

Disponibili inoltre tutti i titoli delle seguenti case:
Aegis,
Berkeley (Geos),
Cinemaware,
Electronic Arts,
IHT Gruppo Editoriale,
Microprose,
Rainbird,
subLOGIC,
Zuma.

BUONO D'ORDINE DA INVIARE A: LAGO DIVISIONE SOFTMAIL, VIA NAPOLEONA 16, 22100 COMO, TEL. (031) 300.174

DESIDERO RICEVERE I SEGUENTI ARTICOLI:
[ ] ADDEBITATE L'IMPORTO SULLA MIA CARTASI NUMERO ____________ SCADENZA ______
[ ] PAGHERO' AL POSTINO IN CONTRASSEGNO

TITOLO DEL PROGRAMMA	COMPUTER	CASSETTA/DISCO	PREZZO

PER ORDINI SUPERIORI A LIT. 50.000 E FINO AL 30/9/88 NON SOMMARE LE SPESE POSTALI DI LIT. 5.000

ORDINE MINIMO LIT. 20.000 (SPESE ESCLUSE) GAZETTE 6/88 TOTALE LIRE ____________

COGNOME E NOME ____________

INDIRIZZO ____________ N. ______

CAP ______ CITTA' ____________ PROV. ___ TELEFONO ____________

FIRMA (SE MINORENNE QUELLA DI UN GENITORE)
VERRANNO EVASI SOLO GLI ORDINI FIRMATI ____________

LA POSTA DELLA GAZETTE

LA VOCE DEI NOSTRI LETTORI

PAL E NTSC

Spettabile Redazione di *Commodore Gazette*, ho letto con grande interesse l'intervista a Dan Silva pubblicata sull'ultimo numero della vostra rivista a proposito della versione italiana di *Deluxe Paint*. Purtroppo però dopo pochi minuti ho provato un senso di crescente delusione. Certo, lui vive in America, e può benissimo non sapere nulla del mercato europeo, ma a quanto pare nessuno si è preoccupato di dirgli che da noi l'Amiga funziona in modo PAL.

Invece gli importatori di software sembrano convinti che la cosa più importante sia leggere "numero dei colori" al posto di "number of colours". O perlomeno questo si deve dedurre dalla foto pubblicata a pagina 36, dove sono in bella evidenza le varie risoluzioni a 200 e 400 linee.

Per quanto tempo i possessori della macchina più dotata del mondo dal punto di vista delle capacità grafiche, dovranno vivere con la sensazione che il loro monitor sia passato sotto lo schiacciasassi?

C'è un'altra cosa che probabilmente Dan Silva ignora, ma che gli importatori nostrani dovrebbero certamente sapere, se vogliono continuare a fare il loro mestiere: presso il mercato pirata esistono già versioni in modo PAL della maggior parte del software esistente. Io stesso possiedo una versione a 256 linee di *Deluxe Paint*, che ho comprato per L. 15.000 dal più importante pirata di Torino (non vi dirò chi è, vi basti sapere che adesso si trova nell'elenco dei Commodore Point). Sono dispostissimo a comprare software originale, ma a condizione di avere un prodotto migliore, e non peggiore, di quello che trovo sul mercato illegale! Se un'operazione del genere la può fare un pirata, e non viene fatta invece dal produttore o dall'importatore ufficiale, allora lo si deve solo a disinteresse per la clientela.

Lo stesso disinteresse per le esigenze normali del mercato europeo si ricava da molti altri prodotti di altre case. Il campo per me di maggior interesse è quello del word processing, e come voi sapete benissimo molti prodotti di questo genere non prevedono le vocali accentate e gli altri caratteri delle più importanti lingue europee. Ma chi spende milioni per comprare computer e stampante, non lo fa certo per scrivere la letterina a Babbo Natale! Un prodotto che pretenda di essere professionale (un discreto esempio è il *Word Perfect* che sto usando ora) deve permettermi di rispettare le regole dell'ortografia italiana, deve permettermi di scrivere in francese o in tedesco quando ne ho bisogno, e così via.

Certo, uno può fregarsene e scrivere *perche'* e *cioe'* al posto di *perché* e *cioè*; uno può trovare il modo di arrangiarsi, ma dev'essere chiaro che l'utente che "si arrangia" non è certo quello che compra software originale!

Maurizio Pistone
Torino

Indirizzate tutta la corrispondenza per la rivista a:

COMMODORE GAZETTE
La posta della Gazette
Via Monte Napoleone, 9
20121 Milano

Preghiamo i lettori di essere concisi e concreti, per darci modo di rispondere al più grande numero possibile di lettere. La redazione si riserva comunque il diritto di sintetizzare le lettere troppo lunghe.

UN MANUALE POCO CHIARO

Spettabile Redazione di *Commodore Gazette*, sono un ragazzo sedicenne, neo-utente dell'Amiga 500 e, dopo essermi congratulato per la stupenda rivista, passo a porvi alcune domande.

1) Quando si inserisce il disco del *Workbench* le prime scritte che appaiono sono il copyright e la versione del *WB*. Quest'ultima può essere sostituita da una qualsiasi stringa modificando la startup sequence del *WB*. Si può fare lo stesso con il copyright o la scritta è inserita nel *Kickstart*?

2) Come posso trasferire il file "AmigaBasic" (contenuto nel disco Extras) sul disco del *WB* senza ottenere il messaggio "Disk Full"?

3) Dopo aver creato un'icona con IconEd come posso sostituirla a una preesistente senza cancellare i file a essa associati?

4) A cosa serve l'icona Empty nel *WB*?

Scusate la mia ignoranza, ma il manuale in dotazione all'A500 non è molto chiaro.

Un lettore
Salerno

1) La modifica del copyright non è possibile agendo sulla startup sequence, in quanto questo messaggio viene visualizzato dal Kickstart.

2) Il disco sistema contiene tanti file che non è possibile memorizzarne altri. Molti di questi file sistema non vengono mai impiegati, come alcuni driver di stampa e fonti carattere. Copiando semplicemente il disco sistema su un altro disco, e cancellando da quest'ultimo i file sistema che non vengono mai impiegati, si riesce a guadagnare lo spazio desiderato. La copia diventa il nuovo disco sistema. Una copia dovrebbe essere fatta sempre, in ogni caso, quando si desidera accedere al disco sistema in scrittura, in modo da preservare l'originale.

3) Quando si ricorre a IconEd, questo avvisa l'utente che le icone visualizzate nei 9 buffer sono di tipo tool, e che se si desidera modificare o sostituire un'icona di diverso tipo, bisogna caricare in uno dei buffer un'icona qualunque di quel tipo, poi copiarla nel buffer principale, e infine alterarla. A disegno ultimato si può memorizzarla su disco assegnandola al file del quale volevamo modificare l'icona, oppure assegnarla a un altro file. Se invece si disegna un'icona di tipo tool, e poi la si assegna a un file dotato di una diversa icona, questo file non può più essere eseguito, a meno che non ne venga sostituita nuovamente l'icona con una di tipo appropriato.

4) Il cassetto EMPTY è una subdirectory vuota che è stata inclusa nel disco sistema per evitare all'utente di dover creare una nuova directory, con i comandi dell'AmigaDOS, ogni volta che ne ha bisogno. In questo modo, infatti, se desideriamo inserire dei cassetti su un disco vuoto, è sufficiente copiare il cassetto EMPTY tante volte quanti sono i cassetti che ci occorrono, cambiando il nome della copia prima di effettuare la successiva.

GEOS 128 E LA STAMPANTE AD AGHI

Sono in possesso della stampante Star NB-15 a 24 aghi e quando uso *GEOS 128* non stampa a 24 aghi ma a 9, ottenendo un risultato di stampa insufficiente. Vorrei pertanto sapere come devo regolarmi per ottenere la stampa a 24 aghi con *GEOS*, facendo presente che sia nel manuale della stampante, sia nel manuale di *GEOS* non ho trovato nulla in proposito.

Gerardo Canoro
Lucca

Nel pacchetto originale di GEOS 128 esiste un driver dedicato alla sua stampante: il driver di stampa Star NB-15. Se questo non dovesse funzionare correttamente, è possibile impiegare il driver di stampa Star NB-15 che viene distribuito in genere nei pacchetti applicativi per GEOS 64, in quanto GEOS 128 e GEOS 64 impiegano lo stesso formato per i driver di stampa. Riguardo al driver Star NB-15 per GEOS 64, la Berkeley fornisce alcuni dati tecnici che forse la possono aiutare. L'interfaccia Centronics dev'essere, se possibile, configurata in modo Epson. Inoltre, dal momento che questa stampante dispone di una risoluzione verticale e orizzontale maggiore di quella impiegata da GEOS, viene sottolineato che le stampe risultano più piccole che con altre stampanti a 8 punti, in modo grafico. Per ovviare a questa riduzione dell'immagine su carta, il driver di stampa dovrebbe realizzare un ingrandimento, associando a ogni pixel di schermo un maggior numero di pixel sulla stampante, sia verticalmente che orizzontalmente. Ma non ci risulta, purtroppo, che esista qualche applicazione destinata a questo scopo.

ASSISTENZA O NON ASSISTENZA?

Ho acquistato nel gennaio '87 un C-128D; ha subito manifestato problemi con la gestione della faccia inferiore del disco, problemi che i quattro interventi in garanzia non hanno eliminato.

Dov'è allora la rinnovata celerità e competenza dei Centri Assistenza Commodore? È tutto qui il servizio post-vendita?

Lo scorso dicembre ho acquistato una MPS1200. Anch'essa, due settimane fa, è stata consegnata all'assistenza. Questa mattina una signorina mi ha gentilmente risposto che la stampante non è ancora pronta e che tutto dipende da quanti ancora attendono prima di me; forse la prossima settimana...

Marco D'Annibale
Oste (FI)

SOFTWARE GALLERY

UNA GUIDA PER ORIENTARSI NEL MONDO DEL SOFTWARE

SHAKESPEARE

Computer: Amiga
Supporto: Disco
Prezzo: L. 250.000
Produzione: Infinity Software
Disponibile presso: Lago (Via Napoleona 16, 22100 Como - 031/300174)

Benché porti il nome del più grande scrittore inglese di tutti i tempi, questo non è un programma di scrittura ma un "integratore di pagina". Il fatto che venga presentato come un "desktop publisher" è solo un incidente di percorso, perché in effetti non possiede le caratteristiche tipiche di questo genere di programmi.

Shakespeare è stato ideato per l'integrazione di testo e grafica sulla stessa pagina; può inoltre far convivere fonti diverse l'una accanto all'altra, consentendo un'ampia varietà di corpi carattere e di colori. E può inviare il risultato a una stampante laser o a colori. E dal momento che i risultati sono a colori, *Shakespeare* rappresenta un notevole passo avanti rispetto ai desktop publisher autentici e agli altri programmi destinati all'integrazione di testo e grafica.

L'elemento base utilizzato da *Shakespeare* è il box, che viene costruito selezionando l'opportuno "strumento" dal toolbox e usandolo per disegnare sullo schermo. All'interno del box può essere collocato un testo o un elemento grafico, ma non tutti e due insieme: per quanto sembri strano, l'integrazione viene realizzata proprio mantenendo tutto separato.

A questo punto dovrebbe essere ovvio che ogni pagina può contenere più di un box, e in effetti non ci sono praticamente limiti al loro numero. Immaginiamo di voler creare una *newsletter*: probabilmente avremo bisogno di un box per la testata, un altro per i titoli, altri ancora per i disegni e, come contorno, diversi box che ci serviranno per racchiudere le colonne di testo.

Per tracciare i box siamo aiutati dalle linee graduate presenti sullo schermo (che possono essere impostate con distanze misurate in pollici o in pica) e dal lento scorrimento che si ottiene quando un box viene esteso oltre i confini visibili dello schermo. In quest'ultimo caso, sullo schermo possiamo vedere solo una parte della pagina, più o meno grande a seconda della risoluzione scelta: per default si ha l'alta risoluzione (640 x 400), ma ci sono anche la media e la bassa risoluzione, e il modo interlace.

Il manuale suggerisce d'iniziare con la media risoluzione e chi scrive si associa al consiglio, per-

SCHEDA CRITICA

INSUFFICIENTE
Un pessimo prodotto che non merita nessuna considerazione.

MEDIOCRE
Il programma ha alcuni difetti di fondo, anche se nel complesso raggiunge quasi la sufficienza.

DISCRETO
Un prodotto accettabile, ma non aspettatevi grandissime emozioni.

BUONO
Raccomandato vivamente: tra i migliori programmi della sua categoria.

OTTIMO
Eccezionale! Fino ad oggi non si era mai visto niente del genere.

ché così si ottiene uno schermo molto simile a quello standard dell'Amiga. E benché testo e disegni, a causa di questa scelta, possano sembrare troppo "compressi" verticalmente, nella stampa si ottengono comunque le giuste proporzioni.

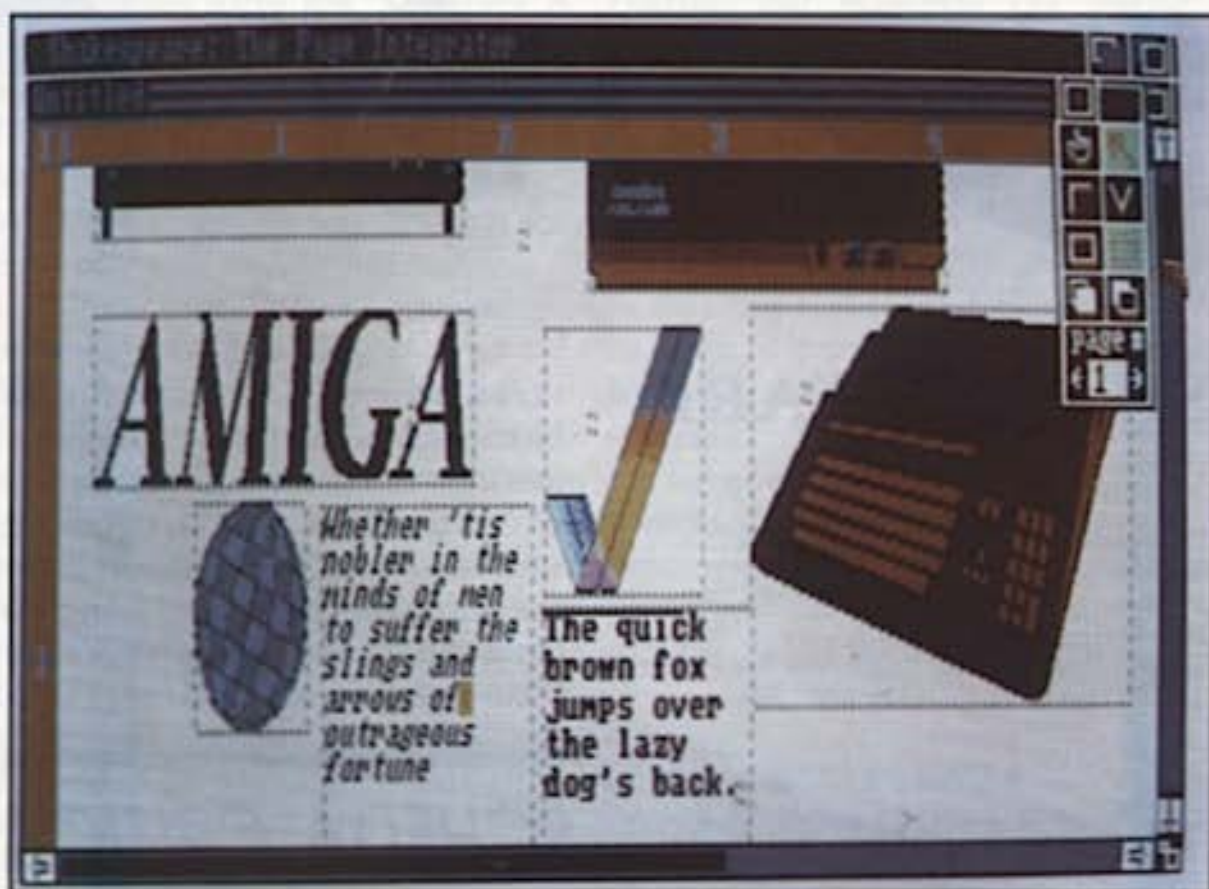
L'aspetto più affascinante di *Shakespeare* è il modo in cui gestisce i file. Supponiamo che il lavoro sia già stato predisposto prima d'iniziare a dar vita alla pagina: è possibile chiamare una directory (per esempio quella delle fonti carattere inclusa in *Shakespeare*) e contrassegnare tutti i file grafici e di testo che si utilizzeranno. Come risultato, la maggior parte del tempo che in genere si spreca per gli accessi al disco può essere risparmiato, e quando si è pronti a ricorrere a una particolare opzione, questa è subito a portata di mano.

Benché i testi possano essere creati direttamente sullo schermo di lavoro di *Shakespeare* nella fonte di default Topaz 8, spesso è più semplice crearli con un buon word processor come *Textcraft*, *Prowrite*, *Scribble*, o qualsiasi altro, purché possa generare file ASCII. Una volta creati, possono essere trasferiti in *Shakespeare* per mezzo dell'opzione Chart presente nei menu a discesa. Per aggiungere poi i file alla pagina si seleziona Text Tool e si apre un box spostando il puntatore del mouse nella posizione voluta.

Nel caso che il box non sia abbastanza grande da contenerlo completamente, il file continua automaticamente nel primo box che verrà aperto subito dopo. Questa caratteristica di continuazione automatica rende inoltre possibile sistemare un disegno sullo schermo e quindi "accerchiarlo" con il testo, aprendo una serie di box sui due lati, sopra e sotto; una caratteristica, tra l'altro, che non molti desktop publisher mettono a disposizione.

Volendo alternare le fonti per ottenere uno stampato più vario, è sufficiente selezionare il blocco di testo che interessa portandovi sopra il puntatore del mouse, e quindi passare al menu a discesa delle fonti e selezionare quella che si desidera usare. Ma mentre la maggior parte dei programmi di questo tipo richiede che per ogni fonte sia stato preparato un box a parte, *Shakespeare* non impone limitazioni di sorta: se il box è abbastanza grande può accogliere tutte le fonti che vogliamo.

Si tratta di una caratteristica utilissima da ogni punto di vista, che mette inoltre a disposizione una possibilità assolutamente unica: la prima lettera di ogni brano di testo può essere variata di colore, fonte e corpo, permettendo di ottenere risultati di un'eleganza che in genere associamo agli artistici manoscritti del medioevo. È un elemento di distinzione che può contribuire a rendere più fantasiosa e personale la nostra pagina.

Nel caso di un progetto più convenzionale, la caratteristica delle "fonti miste" ci permette di creare nella nostra ipotetica *newsletter* la testata della pagina, la data, il copyright e il titolo entro i confini di uno stesso box, e ciascuna riga può avere caratteri e dimensioni diverse, ed essere giustificata per suo conto. Baste-

Alcuni esempi di testo, grafica e fonti carattere diverse su una stessa pagina

A ogni box è associata una particolare palette. Sullo schermo i colori dei disegni dipendono dal box selezionato, ma le anomalie scompaiono nella fase di stampa

rebbe questa caratteristica, di cui insistiamo a sottolineare la fondamentale importanza, per capire che grande passo avanti sia *Shakespeare*... chiedetelo a chiunque abbia usato un altro integratore di pagina. Per l'utente avere questa possibilità rende molto più pratica e veloce la fase di realizzazione della pagina.

Dal momento che mette a disposizione il formato grafico IFF, *Shakespeare* è in grado di accettare disegni e diagrammi creati per mezzo di programmi come *Deluxe Paint II, Aegis Images, Photon Paint* e molti altri. Per mantenere i colori con cui l'immagine è stata creata, ciascun box della pagina è abbinato a una particolare palette. Di conseguenza, a volte capita che tutti i box (fuorché quello selezionato) mettano in mostra assortimenti di colori davvero bizzarri: d'altra parte, come capita con le distorsioni che risultano dall'uso di certe risoluzioni di schermo, tutto va a posto quando si passa alla stampa e ogni elemento della pagina viene stampato tenendo conto della sua reale palette.

Shakespeare avrà un nuovo set di driver di stampa creati sulla base del *Workbench 1.3*, i quali al momento della stesura di questo articolo non sono ancora disponibili. Quando si ricorre a Preferences per una variazione della stampante, si noteranno due box Graphic Select, contrassegnati 1 e 2; il primo è quello consueto, dove si selezionano i parametri aspect, shading, threshold, black and white. Il secondo prevede regolazioni fini come anti-aliasing, dithering, density e scaling. Poiché queste ultime caratteristiche sono tipiche più del *Workbench 1.3* che dello stesso *Shakespeare*, non c'è documentazione che spieghi come servirsene: finché non sarà disponibile il manuale del nuovo *Workbench* è opportuno che vengano lasciate con i loro valori di default.

Questi nuovi driver, in ogni caso, riescono a ottenere una stampa a colori molto più pulita di quanto si possa credere, almeno con una stampante Okimate 20.

Inoltre, *Shakespeare* fornisce un secondo set di Preferences, selezionabile da un menu a discesa, che contempla voci come la grandezza della pagina e la quantità di punti per pollice. Se la nostra stampante è presente nelle Preferences, e la selezioniamo, allora l'impostazione di default risulta essere automaticamente quella corretta.

Una volta che la stampa ha effettivamente inizio, un requester permette d'impostare velocità di stampa e numero di copie.

Oltre al supporto per tutte le tradizionali stampanti presenti in Preferences, viene fornito supporto anche alle stampanti laser in grado di "leggere" il linguaggio PostScript di descrizione della pagina. L'impostazione di default è per la LaserWriter Plus della Apple, ma possono essere installati facilmente altri driver per stampanti di tipo PostScript. Nel disco delle fonti è incluso anche un file di fonti carattere speciali per la LaserWriter della Apple.

Quando la pagina raggiunge finalmente il suo aspetto definitivo, scopriamo che ci sono due modi per salvare un file. Si può salvare (o caricare) il file nel modo "issue", che salva l'intera pagina con tutti i grafici, disegni e variazioni di fonti carattere che sono state usate, comprese le informazioni sul layout.

Il modo "article", invece, salva solo il box selezionato, di testo o di grafica, e di conseguenza non salva nessuna informazione sulle

relazioni che legano il box al layout. Questo è sottinteso dallo stesso nome dell'opzione, in quanto, in un certo senso, un *articolo* fa parte a sé e non ha una stretta relazione con gli altri elementi della pagina.

La documentazione fornita assieme a *Shakespeare* è senz'altro esauriente. Gradino per gradino, guida l'utente attraverso la stesura della pagina fino alla stampa del primo documento, prendendo poi in considerazione le tecniche più avanzate e concludendo con un'utile serie di appendici e un indice.

Inoltre *Shakespeare* è copiabile, e può quindi essere installato su hard disk. Benché il programma giri con 512K e un solo drive, secondo noi l'optimum è avere a disposizione 1 MB di RAM e almeno un drive aggiuntivo: le directory delle fonti sono troppo estese per essere incluse nel disco programma, e un secondo drive per il disco delle fonti minimizza i tempi morti.

Per concludere, benché manchi di alcune finezze per l'impostazione della pagina di testo (come la possibilità di variare interlettera e interparola) che sono indispensabili a qualunque desktop publisher, e benché l'edit di schermo sia lento (perché i testi vengono trattati come la grafica), *Shakespeare* è comunque molto vicino al desktop publishing vero e proprio; fornisce infatti un output sia a colori sia in bianco e nero ed è in grado di trasmettere a una stampante laser le pagine che crea.

Tutto questo, e la capacità di manipolare i box di testo con molta elasticità, fa di *Shakespeare* un programma che dovrebbe essere preso seriamente in considerazione da chiunque sia interessato a realizzare *newsletter*, opuscoli, lettere personalizzate, bollettini, e così via, ottenendo un risultato professionale.

E.B.

VIDEOTITLER

Computer: Amiga
Supporto: Disco
Prezzo: L. 199.000
Produzione: Aegis
Disponibile presso: Lago (Via Napoleona 16, 22100 Como - 031/300174)

Possedere un videogeneratore elettronico di testo richiedeva fino a qualche tempo fa l'impiego di sistemi costosi e molto sofisticati. L'Amiga, però, ha fatto il suo ingresso anche in questo settore, e ha rivoluzionato il sistema di valori preesistente, fornendo i mezzi per produrre a un prezzo più che abbordabile complesse e raffinate creazioni di computergrafica. È stato un grosso passo avanti nella rivoluzione del desktop video.

VideoTitler, una recente produzione della Aegis Development, è uno dei programmi che si inseriscono in questo filone. Si tratta di un sistema in grado di generare titoli e slide show di alta qualità, che sfrutta a fondo tutte le possibilità dell'Amiga e la sua ben nota semplicità d'uso.

L'espressione "sfruttare a fondo" va intesa nel senso letterale del termine; il programma, infatti, necessita di almeno 1 MB di memoria RAM per lavorare in bassa risoluzione (320 x 200) o in risoluzione video (320 x 400), e di almeno 1,5 MB per la media e alta risoluzione. Sono particolari che già a un primo sguardo evidenziano il carattere professionale di questa stazione di lavoro, che non si presta certo a essere utilizzata come un divertente giocattolo... o almeno, non solo per quello.

All'interno del disco sistema, venduto in un'elegante confezione corredata di manuali in inglese completi e dettagliati, si trova anche *VideoSEG* (Video Special Effects Generator, generatore di effetti speciali su video), un programma che completa efficacemente *VideoTitler* e che può essere usato anche con altre applicazioni grafiche e d'animazione.

Veniamo ora alle caratteristiche generali. Come già abbiamo detto, *VideoTitler* è in grado di lavorare (se la quantità di memoria disponibile lo permette) in quattro risoluzioni. Il programma può girare sia in NTSC sia in PAL, e creare titoli in ciascuno di questi due formati. Le varie funzioni si utilizzano tramite menu a scomparsa e tastiera; per quest'ultima viene fornito un cartoncino di riferimento dei comandi.

Per il testo, come c'era da aspettarsi, si può utilizzare qualsiasi fonte carattere, purché creata nel formato IFF; per esempio si possono usare le fonti a colori create con *The Calligrapher*, un programma della InterActive Softworks. C'è inoltre la possibilità di accedere alle cosiddette "poly font", create appositamente per questo programma, che possono essere inclinate, rovesciate, distorte, ingrandite o rimpicciolite a piacere con semplici movimenti del mouse.

Per ogni fonte poi si possono

scegliere venti stili diversi per realizzare "effetti speciali" o si può addirittura creare uno stile proprio utilizzando il modo operativo "Expert mode". A tutte le schermate può essere applicato l'overscan (a tre livelli: off, medium e severe); si tratta di una funzione che consente di trasferire schermate su un videoregistratore o un monitor eliminando la

presenza del bordo.

Probabilmente la caratteristica più interessante del programma è la possibilità di utilizzare immagini, finestre e pennelli in formato IFF, per costruire sfondi delle schermate contenenti i titoli. Qualsiasi immagine creata con un programma che generi file IFF può essere caricata in memoria e inserita nella cosiddetta "clipboard", che tramite funzioni di *cut and paste* permette di modificare forma e dimensioni dell'immagine prima d'inserirla nella schermata. Tutte le immagini IFF, inoltre, possono essere salvate su disco per essere utilizzate in un secondo tempo con programmi di disegno e animazione o con *VideoSEG*.

Abbiamo parlato finora delle opzioni più originali, ma *VideoTitler* offre anche le tipiche caratteristiche dei migliori programmi di disegno. Per esempio è possibile creare una propria palette scegliendo i colori tramite uno speciale "color requester" detto HAM (Hold And Modify). Si tratta di una barra che appare nella parte bassa dello schermo e che contiene tutti i famosi 4096 colori di cui è dotato l'Amiga: muovendosi lungo la barra con il mouse è possibile scegliere direttamente il colore e la sfumatura preferita, senza dover agire sulle intensità di rosso, verde e blu. Le palette con cui lavora il programma contengono 32 colori (in risoluzione bassa o video) e 16 colori (in media e alta risoluzione). Inoltre, se si possiede un A500, un A2000 o un modello recente di A1000, *VideoTitler* può sfruttare il modo "Extra Halfbrite", che è in grado di raddoppiare il numero di colori disponibili in bassa risoluzione (e video) creando una seconda palette con colori dotati di intensità e saturazione dimezzate. Il vecchio A1000 non dà supporto a questo modo grafico ma, volendo, si può sempre rimpiazzare il proprio chip "Paula" di vecchio tipo con quello nuovo.

Con almeno 1,5 MB di memoria, infine, è possibile animare i propri titoli utilizzando il formato Aegis/Sparta ANIM. Si tratta di un formato (da utilizzare con il programma *VideoSEG*) che "comprime" i file immagazzinando solo le differenze fra le varie immagini, e non le immagini complete, riuscendo a ridurre notevolmente le dimensioni dei file.

Per finire, tramite *VideoSEG* si possono collegare immagini diverse, organizzando slide show con vari effetti di animazione e dissolvenza, a completamento di un lavoro che sarà quasi all'altezza delle realizzazioni di stazioni grafiche professionali, come quelle utilizzate in ambito televisivo e pubblicitario.

N.F.R.

CORPORATION

Computer: C-64/128
Supporto: Cassetta/Disco
Prezzo: L. 18.000/25.000
Produzione: Activision
Disponibile presso: Lago (Via Napoleona 16, 22100 Como - 031/300174)

Nell'anno 3026 non c'è più traccia di confini e di Stati sovrani: ogni identità nazionale è stata soppiantata dall'appartenenza a una delle onnipotenti Corporazioni.

In questo scenario da incubo è stato ambientato *Corporation*, un gioco di strategia e d'azione recentemente immesso sul mercato dalla Activision. Il giocatore veste i panni del comandante di una squadra mineraria robotizzata al servizio di una Corporazione. Ha il compito di ricercare ed estrarre dal suolo di un asteroide la maggior quantità possibile del preziosissimo cristallo Minorthiano.

Le principali difficoltà che si frappongono all'adempimento della missione sono due. Innanzitutto l'asteroide sul quale si trova il cristallo ha una vita molto limitata; in poche ore, infatti, entrerà in collisione con una luna vagante e si disintegrerà. Un'altra unità di estrazione, inoltre, appartenente a una Corporazione rivale, giungerà sull'asteroide contemporaneamente al giocatore, e tenterà con ogni mezzo di ostacolarlo e di portare via quanto più cristallo possibile. È ovvio che solo una delle due spedizioni potrà ottenere il successo sperato.

Dopo aver caricato il gioco, compare uno schermo da cui si possono ottenere informazioni sulla missione e sul tipo di MRU (unità mobili robotizzate) a disposizione. Successivamente compare la pagina per selezionare i mezzi, il numero dei giocatori e la durata della missione; i mezzi (di trasporto, di attacco, di ricerca e scavo...) fondamentalmente sono di due tipi: a energia solare e a fusione nucleare. I primi sono meno costosi ma si fermano nel momento in cui arrivano nella parte in ombra dell'asteroide, i secondi sono sempre in grado di lavorare ma sono molto costosi. La selezione del giusto rapporto fra gli uni e gli altri è fondamentale per la buona riuscita del gioco.

Dopo aver fatto le dovute scelte, l'unità di estrazione mineraria atterra sull'asteroide e il gioco prende subito avvio. Per gestire le MRU si utilizza il joystick, con il quale si segnala alle varie unità il punto che devono raggiungere e il percorso che devono compiere. Ricevuti gli ordini, le unità si muoveranno automaticamente lasciando il giocatore libero di concentrarsi su altre fasi del gioco. Lo schermo è diviso in due parti uguali, una per il giocatore e l'altra per il computer o il secondo giocatore. Nella parte alta degli schermi si trova una griglia che rappresenta la superficie dell'asteroide, in cui si può vedere la posizione della propria base e il lento movimento della zona d'ombra.

La parte centrale dello schermo contiene invece la visualizzazione dell'azione vera e propria: in questo riquadro viene riprodotto un tratto di superficie corrispondente a un quadrato della griglia superiore; è qui che si impostano i comandi per le MRU, si controllano i giacimenti che sono stati scoperti e si programma l'estrazione del cristallo minorthiano. Questo schermo inoltre si trasforma in una vista frontale da un carro se si effettuano azioni di disturbo contro i rivali con carri lanciarazzi. L'azione resta comunque molto limitata e statica. In fondo tutta l'abilità del giocatore sta nell'effettuare le giuste scelte dei mezzi e nella velocità con cui imposta i comandi alle MRU che, però, poi si muovono da sole.

La grafica del programma è ben realizzata, ma la schermata divisa in due non fa sicuramente un bell'effetto. Il gioco è poco elaborato anche dal punto di vista strategico: si devono estrarre cristalli il più in fretta possibile, senza perdere troppo tempo in azioni di disturbo contro il "nemico", che ci darebbero ben poco vantaggio....

In definitiva, se vi regalano *Corporation* non buttatelo via, ma non aspettatevi neanche troppe emozioni.

N.F.

FERRARI FORMULA ONE

Computer: Amiga
Supporto: Disco
Prezzo: L. 69.000
Produzione: Electronic Arts
Disponibile presso: Lago (Via Napoleona 16, 22100 Como - 031/300174)

La Electronic Arts ha ormai abituato la sua clientela a prodotti davvero pregevoli. Un ulteriore esempio è questo drive simulator, che però è ancora ben lontano dai desideri di quanti vorrebbero realmente guidare un bolide da corsa. Spiacente di deludervi così, fin dalle prime righe, ma la realtà è che al giorno d'oggi è sempre più diffusa l'abitudine di vendere fumo, magari di ottima qualità ma pur sempre fumo. È in questi casi che diventa

evidente il valore di una recensione: 99 persone su 100 sicuramente acquisterebbero il programma fidandosi delle immagini di copertina... che sono effettivamente di ottima qualità, ma servono più che altro a nascondere la pecca principale del programma: la scarsa giocabilità. Intendiamoci, dalla grafica all'azione, alla preparazione della vettura, è tutto stupefacente. Siamo di fronte a un'accuratezza davvero esemplare. Purtroppo poi arriva il momento di guidare, e l'entusiasmo cala rapidamente. Chiusa questa lunga ma doverosa premessa, passiamo a esaminare i dettagli del gioco.

Un momento fondamentale di ogni partita è sicuramente la messa a punto della propria F186: i box di Fiorano, ipotetico punto d'arrivo dei meccanici di tutto il mondo, fanno da sfondo a tutto il lavoro di preparazione. Senza soffermarci troppo, diremo che la vettura verrà sottoposta a una serie di controlli e regolazioni della meccanica che influiranno in modo determinante sul suo comportamento in gara. Per i più esigenti è bene aggiungere che è previsto il collaudo del motore, addirittura con una simulazione del suo funzionamento in tempo reale (compreso l'effetto del turbo), tenendo conto che pressioni, voltaggi e temperature sono fondamentali per un buon set-up della vettura. Un giro di collaudo potrebbe essere un ottimo sistema per arrivare a una corretta messa a punto, ma naturalmente una volta trovati i giusti "valori" della vettura, tutto il rito della preparazione non avrebbe più senso. Dimenticavo: nel giro di qualificazione, così come durante la gara, è fondamentale ricordarsi di fare il pieno (inutile ridere, il più delle volte la vettura torna ai box spinta a braccia!).

Per trasferirsi da Fiorano al circuito scelto per la prima gara del Grand Prix, la vettura viene

caricata su un camion con l'effigie del cavallino rampante, e parte per un viaggio aereo durante il quale vediamo suggestivi panorami del nostro pianeta. Benissimo, a tutti noi fa piacere vedere che esistono programmatori perfezionisti, ma la felicità è di breve durata perché sta per iniziare la gara vera e propria. Dopo un'ultima messa a punto della vettura nel box del circuito, ci si trova improvvisamente catapultati nell'abitacolo (e qui non c'è proprio nessun tentativo di far vedere il pilota che si prepara, si infila il casco e magari saluta la folla). Tre, due, uno.... ma non parte! Ebbene sì, a volte la vettura non parte, e le telefonate alla EA o a Maranello non servono a risolvere il problema.

Se si riesce a partire, si ha la sorpresa di verificare che il mouse non serve solo nelle fasi preliminari, ma anche per guidare la vettura! E questo, sono lieto di ammetterlo, è un grosso vantaggio. Sinistra, destra, e pulsanti per accelerare e frenare; il cambio, che può essere automatico, è visibile a bordo. Vengono indicati la velocità e il carburante, e sono presenti gli specchietti retrovisori, fondamentali per la guida. Sorpassi, incidenti (purtroppo senza alcun effetto scenico) sono simili a quelli di un altro grande della simulazione automobilistica, *Revs*, e anche qui riuscire a completare la gara (magari arrivando ultimi, ma almeno prima che il computer si fonda) è un'azione da veri eroi mouserizzati.

Il confronto tra gli analoghi prodotti esistenti in commercio può dare un solo responso: *Ferrari Formula one* è incontestabilmente al primo posto, ma è ancora ben lontano – per giocabilità e controllo della vettura – da quanto noi, comuni mortali, attendiamo vanamente ormai da mesi. La parola d'ordine è ancora "tutto ciò che fa scena si vende bene". *Test drive* in testa. **L.M.**

GEOS 117

Computer: C-64/128
Supporto: Disco
Prezzo: L. 14.000
Produzione: Pizza Software
Distribuzione: Adventura (in tutte le edicole)

Sedici personaggi, più di 2.500 locazioni visitabili, oltre 60 schermate grafiche: questi i dati strutturali della bella avventura disponibile in tutte le edicole grazie alla nuova rivista-disco *Adventura*.

Il genere è quello del giallo fantascientifico, un connubio che ha attratto i migliori autori sia dell'uno che dell'altro genere. Citiamo almeno il nome ormai classico di Isaac Asimov, con i suoi romanzi *Abissi d'acciaio, Il sole nudo, I robot dell'alba*, i suoi investigatori robotici, e assassini che possono nascondersi in una breccia temporale...

È in quest'area dell'immaginario che trova la sua collocazione *GEOS 117*, l'avventura grafica prodotta dalla Pizza Software e distribuita con il periodico mensile *Adventura*.

I "georeattori" sono i pilastri del mondo di questo gioco. Intorno a loro si sono organizzate le comunità agroindustriali in cui vive l'umanità. Un mondo idilliaco, immerso nel verde, essenzialmente pacifico. Eppure... piccole crepe lasciano intravedere i segni di una violenza ancora non sopita.

Proprio a GEOS 117, all'inizio del secondo turno di lavoro (ogni turno dura 72 ore), scompare inspiegabilmente McClaud, uno dei 16 tecnici addetti al controllo del georeattore della comunità.

Due giorni dopo un altro tecnico, Thomas, è vittima di un inspiegabile incidente. La situazione è particolarmente tesa, anche perché, fin dal momento dell'attivazione del georeattore, le misure di sicurezza avevano evitato il benché minimo inconveniente. La passerella da cui Thomas è caduto, inoltre, come tutte le attrezzature situate nel livello inferiore del georeattore, risultava in perfetta efficienza.

All'inizio del gioco, quando veniamo informati dell'accaduto, ci troviamo nei panni di Zoltan, un agente investigativo al servizio del Consiglio dei saggi della comunità. Ci viene affidato il caso, e di conseguenza ci vengono forniti un nome di copertura, il codice d'identificazione necessario per accedere alle aree protette del georeattore e altre informazioni di base. Possiamo così dare inizio alle indagini.

Dinanzi a noi si schiude il mondo di *GEOS 117*. Innanzitutto la grande area verde in cui è immersa la comunità, poi gli edifici del personale e infine il piano

inferiore dello stesso georeattore.

Starà a noi rintracciare e interrogare il personale della base. La nostra qualifica di copertura, ispettori alla sicurezza degli impianti, ci garantisce la collaborazione generale. Quanto alla veridicità delle dichiarazioni che otterremo... saranno le nostre capacità professionali a suggerirci se diffidare. Anzi, saranno proprio le eventuali contraddizioni a metterci sulla strada giusta.

Raggiunta una ragionevole

ipotesi di colpevolezza e identificata l'arma del delitto, se di delitto si tratta, non avremo che da attivare la procedura di arresto. Sono stati allestiti tre terminali, a questo scopo: dovremo fornire loro il nostro codice d'identificazione, i dati relativi al tipo di procedura e al tipo di reato ipotizzato, la matricola d'identificazione del colpevole e il nome dell'arma del delitto.

Queste, a grandi linee, sono le premesse e la trama del gioco. Quanto ai suoi elementi costitutivi, non possiamo che elogiarne la grafica raffinata. Come nel precedente lavoro di questa software house, *John Brenner, Boston Celtic*, la fonte d'ispirazione rimane il fumetto. Dai classici del fumetto americano degli anni Cinquanta, questa volta si passa ai fumetti di fantascienza francesi degli anni Settanta, opportunamente rivisitati. Un'attenzione certosina per i particolari, un sapiente dosaggio del colore e le inquadrature di taglio cinematografico pongono il gioco in dichiarata concorrenza con il meglio della produzione straniera.

Interessante è anche la soluzione scelta per il sistema di comunicazione. In alto a destra una piccola finestra informa se al giocatore è richiesto l'uso del joystick e della tastiera, o solo di quest'ultima.

Poco più in basso, la finestra grafica e il logo del gioco sovrastano lo spazio dedicato alle comunicazioni di input e output.

Una nota a parte merita il parser. Dobbiamo prima di tutto evidenziarne la velocità di lavoro, sorprendente in particolare se confrontata con quella dei prodotti più diffusi. L'ampiezza che viene messa a disposizione, 34 caratteri, permette la formulazione di frasi abbastanza elaborate, che comunque non sono quasi mai necessarie. I messaggi di sistema sono sintetici ma chiari. La sigla "n.co." ci avverte che il programma non contiene nel suo vocabolario le parole impiegate. La sigla "n.ca." ci comunica invece che abbiamo usato parole conosciute dal gioco, ma che in quella particolare situazione non hanno senso o non servono.

Anche noi ci diamo all'investigazione, nello spirito di *GEOS 117*, e proviamo a indovinare qualcosa dei progetti futuri di questa software house. Ecco un indizio: mentre ci muoviamo con il joystick esplorando il vasto parco, il rapido alternarsi delle schermate crea un gradevole effetto di animazione che ben si sposa all'impianto generale del gioco. Speriamo che questo primo momento di animazione non resti senza seguito nei prossimi lavori.

"Ultima viene la morte". Questo temuto personaggio è di casa in *GEOS 117*. Un'opportuna utility permette al giocatore di salvare in ogni momento la posizione raggiunta. Ne consigliamo un uso attento. Abbiamo diverse possibilità infatti di "morire". Una morte con ignominia ci attende se violiamo lo spirito del gioco, ad esempio non accettando la missione assegnataci. Moriamo per incidente se ignoriamo i limiti della zona da esplorare o ci avventuriamo incautamente in aperture protette da congegni di sicurezza. Moriamo infine per nostra incapacità se non siamo in grado di risolvere il caso. D'altronde il futuro di *GEOS 117* è nelle nostre mani...

F.T.

RETURN TO ATLANTIS

Computer: Amiga
Supporto: Disco
Prezzo: nc
Produzione: Electronic Arts
Disponibile presso: Lago (Via Napoleona 16, 22100 Como - 031/300174)

Lunghe immersioni nelle profondità oceaniche, avventurose scorribande tra gli esseri marini più singolari, tesori sommersi da recuperare... un intero ecosistema da difendere dalle insidie di speculatori senza scrupoli... Tutto questo (ma non solo questo) ci offre la bella avventura della Electronic Arts che presentiamo in queste pagine.

Vediamo dapprima la cornice in cui si colloca la vicenda. Il mondo di *Return to Atlantis* si trova lungo la nostra linea temporale, in un futuro non troppo remoto. Le risorse del terreno stanno esaurendosi e l'uomo si è rivolto al mare. Un aspro dissidio separa chi vorrebbe dar vita nel profondo degli oceani a una nuova società rispettosa degli equilibri naturali, e chi invece mette al primo posto le spietate leggi del profitto.

Per evitare che questo dissidio sfoci in una vera e propria guerra, un'organizzazione internazionale, la Fondazione (i fan di Asimov non si scandalizzino...) opera per il mantenimento degli equilibri esistenti.

Tentativi di sabotaggio alle comunità marine già esistenti, installazioni di strumenti dannosi per le specie più delicate, cavi sottomarini interrotti: questi sono alcune delle 14 situazioni che il giocatore deve fronteggiare.

Tutto ha inizio a Isla Perdida, la base oceanica della Fondazione. Qui il giocatore riceve le prime informazioni relative alla sua missione e può procedere alla personalizzazione del proprio eroe. Le caratteristiche in gioco sono sette, tra le quali una sola è modificabile.

Poi, secondo i canoni ormai fissati da un'intera tradizione narrativa, al Sea Thief Cafe il giocatore potrà – se ne sarà capace – accedere alle altre informazioni a lui necessarie.

Strani avventori frequentano il locale e ci si deve destreggiare attentamente tra modestia, seduzione e intimidazione per entrare in possesso dei molti brandelli di verità che solo a loro si possono strappare.

Finalmente la missione ha inizio. La *Viceroy*, la bellissima nave della Fondazione, è attrezzata per ogni evenienza. ART, l'Advanced Robotic Transcoordinator, manterrà i contatti con il mondo esterno e con la base; la Medical Beam, se necessario, reintegrerà le energie spese e RUF, il Remote Underwater Friend, accompagnerà il protagonista nelle profondità marine.

A immersione avvenuta, sullo schermo scorrono i fondali oceanici brulicanti di vita. Ogni anfratto può celare un'insidia, ogni roccia un nemico... sta al giocatore scegliere la strategia migliore per raggiungere l'obiettivo della missione, operando in stretta cooperazione con la nave e con il suo aiutante robotico, e servendosi delle attrezzature in suo possesso nel modo più utile ed efficace.

Ma vediamo un po' più nei dettagli, ora che ci siamo familiarizzati con l'ambiente del gioco, in che cosa consiste la prima missione. Il galeone spagnolo *Tristeza*, naufragato nei pressi di Haiti nel 1579, è stato ritr

dagli esploratori. Dovrebbe avere a bordo una cassa piena d'oro proveniente dal tesoro degli Incas. Le coordinate del galeone vengono allora trasmesse a ART a bordo della *Viceroy*. Gli ordini sono: individuare la posizione esatta del tesoro e recuperarlo per poterlo esporre nei musei della superficie, prima che venga raggiunto da qualcun altro, e venduto al mercato nero.

Il giocatore si trova davanti allo schermo, in attesa del primo imprevisto: nella parte superiore vede scorrere un'accettabile simulazione delle sue imprese sottomarine, mentre nella parte inferiore ha a disposizione l'Underwater Actions Menu.

Torniamo ai dettagli tecnici del gioco. L'opzione Swim serve per il movimento del personaggio; si usa invece Scan per ottenere informazioni sugli oggetti ritrovati, Armed per attivare un micidiale sistema d'arma e infine Surface per rientrare alla *Viceroy*. Sempre in questo settore dello schermo figurano quattro icone di segnalazione che il giocatore farà bene a tenere sempre sotto stretto controllo: Air, con l'indicazione della riserva d'aria ancora disponibile nelle bombole; Health, sensibile sia agli esiti dei combattimenti intrapresi sia agli effetti deleteri di ambienti marini tossici; Energy, con i dati relativi alle batterie collegate a eventuali propulsori e infine Pos, con la posizione del protagonista espressa in coordinate cartesiane.

Per concludere, vorremmo mettere in evidenza come questo gioco sia un interessante e riuscito tentativo di fondere insieme simulazione e gioco di ruolo. La Electronic Arts si muove con successo lungo una direzione nuova, ed è su questa strada che dovremo attenderci in futuro i prodotti più interessanti. A tutti diamo comunque appuntamento al Sea Thief Cafe di Isla Perdida.

F.T.

RIMRUNNER

Computer: C-64/128
Supporto: Cassetta/Disco
Prezzo: L. 18.000/25.000
Produzione: Palace Software
Distribuzione: Lago (Via Napoleona 16, 22100 Como - 031/300174)

Realizzato dai programmatori di *Cauldron* e *Barbarian*, i massimi successi della Palace Software, *Rimrunner* si presenta come un prodotto dal soggetto originale, ma con uno svolgimento ormai obsoleto e, purtroppo, non controbilanciato dall'accuratezza dei particolari. Nulla di nuovo quindi, ma anche se la dinamica del gioco ricorda lontanamente l'ormai mitico *Defender* vi sono però alcuni spunti che rendono piacevole fare due o tre partite con *Rimrunner*.

Il nostro personaggio è un "insettoide a cavallo di un runner" che, traducendo dalla lingua che si parla nel pianeta dei programmatori, non è altro che una formica a cavalcioni di un dinosauro modello baby (originale, vero?). Qualche volta c'è da chiedersi se i progettisti di certi giochi non paghino qualcuno (sempre la stessa persona) per la creazione di nomi bizzarri... o perlomeno strampalati.

Nel gioco abbiamo il compito di tenere sempre alto il livello di energia di cinque generatori, unica difesa del pianeta contro nemici non meglio identificati (formichieri spaziali?) che entrano nello schermo in numero sempre maggiore con il decrescere dell'energia. La potenza dei cinque generatori viene indicata in una minimappa posta nella parte superiore dello schermo, e il loro lampeggiare progressivo indica una richiesta d'intervento. Il tutto, naturalmente, scandito dall'inesorabile decrescere di un timer.

Il paesaggio, tipico degli arcade spaziali, è composto da piante e montagne stranissime, ed è affollato da una moltitudine di nemici da cui ci si dovrà difendere, per mezzo di un fucile, durante tutto il corso della partita.

Lo scroll, rigorosamente orizzontale, chiude "a cerchio" la pista di gioco, e risulta perfino troppo facile colpire i nemici sparando in corsa.

In pratica si deve semplicemente correre da un generatore all'altro per ricaricare quelli che stanno per esaurirsi, sparando a quanti più nemici è possibile ed evitando di essere a nostra volta colpiti e uccisi. La parte forse più originale e divertente dell'intero gioco è il rapporto esistente tra la formica e il suo mezzo di locomozione: mentre la prima, se colpita più volte, "morirà" solo teoricamente, facendo cioè decrescere il numero di vite in possesso del giocatore, la morte del secondo è una vera e propria sceneggiata, che va dall'accasciarsi lento al suolo, al fischio della formica che chiama un nuovo "Dino". E sul gioco non c'è proprio nient'altro da dire. Tutto qui.

Certo, l'atmosfera è delle migliori, così come gli effetti e le scelte grafico-sonore, ma, al giorno d'oggi le esigenze del giocatore medio sono orientate verso livelli più alti. Peccato, ma in fondo ci si può divertire anche con così poco.

L.M.

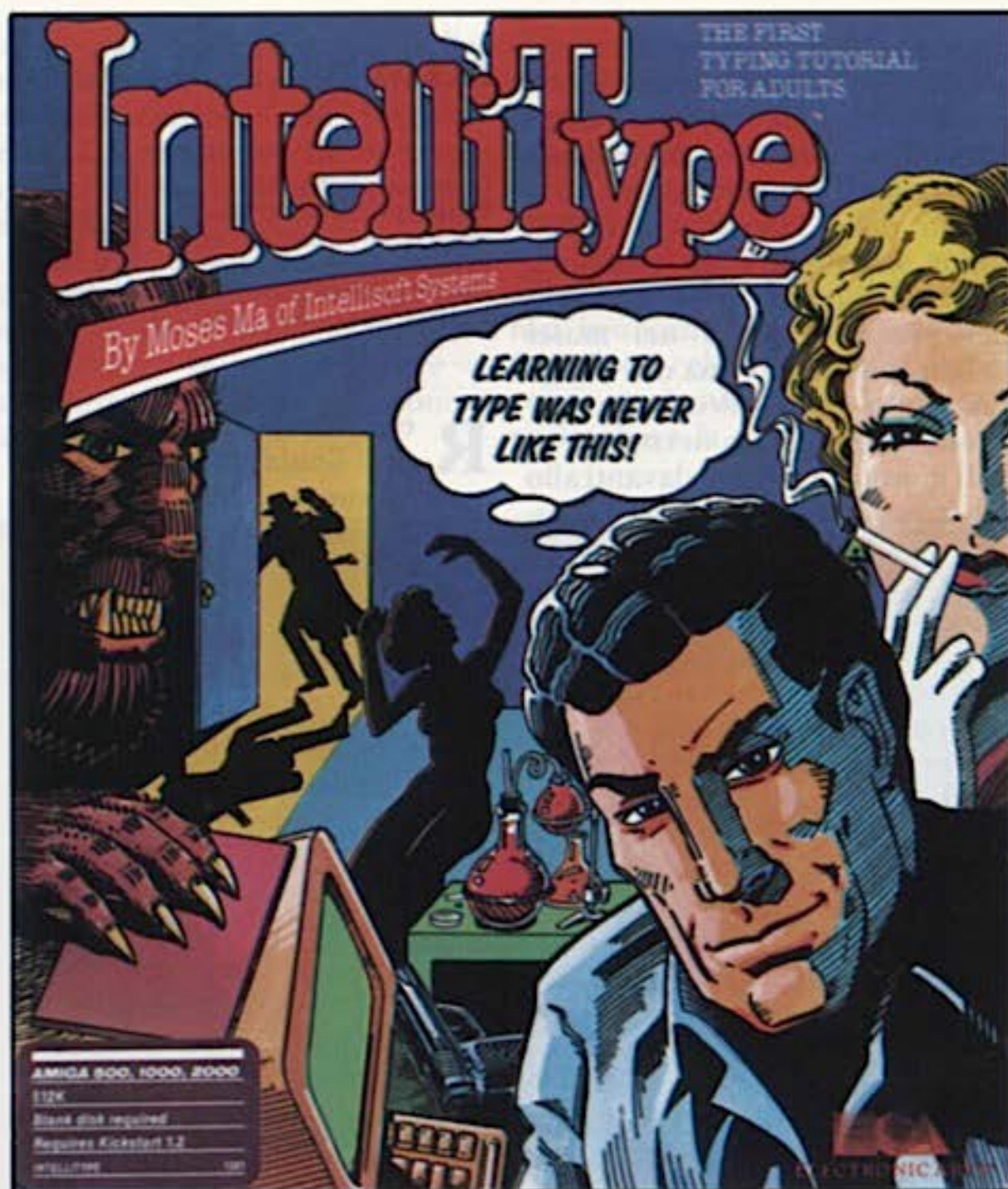
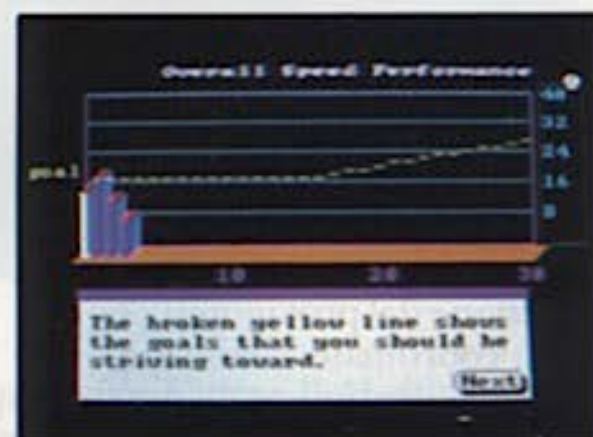
INTELLITYPE

Computer: Amiga
Supporto: Disco
Prezzo: nc
Produzione: Intellisoft
Distribuzione: Electronic Arts (Langley Business Centre 11-49, Station Road, Langley - Nr. Slough Berkshire SL38YN, England)

Al caricamento del programma appare la scritta "typing is fun" (battere a macchina è divertente). Illusione pubblicitaria? Falsa promessa? Niente di tutto questo: imparare a digitare rapidamente i testi, con *Intellitype* è veramente facile e divertente.

Intellitype è un programma di autoapprendimento delle tecniche di digitazione prodotto per l'Amiga dalla Electronic Arts; si tratta di un sofisticato sistema didattico che combina la tecnologia computerizzata, e i principi didattici della dattilografia, con un eccitante racconto d'amore e d'avventura. Mentre si segue la trama della vicenda di Ted e Laura, nel loro ambiente strano e misterioso, si prende confidenza con la tastiera e si migliora la propria velocità e precisione.

Al temine del corso, della durata di 30 giorni o di 60 (un giorno sì e uno no), si dovrebbe, secondo le previsioni dei programmatori, raggiungere una velocità di circa 30 parole al minuto, ma anche se non si riesce a raggiungere questi standard, l'esperienza che si può trarre dal corso sarà preziosa in ogni caso, e migliorerà comunque le prestazioni dei dattilografi meno esperti. Altra qualità non secondaria, le lezioni sono di durata piuttosto breve (30 minuti circa), e non dovrebbero creare problemi di tempo neanche alle persone più impegnate.

Il programma è dotato di caratteristiche particolari che meritano una descrizione un po' più accurata. Innanzitutto, come ho già detto, gli esercizi seguono la falsariga di un romanzo interessante e ben fatto, il che aiuta lo studente a non abbandonare il corso per noia o demoralizzazione... in ogni caso prevale il desiderio di sapere come andrà a finire l'avventura. In secondo luogo, *Intellitype* si basa sui più moderni criteri didattici per la dattilografia: partendo dai fondamenti è in grado di dare un'impostazione corretta anche a chi da anni digita servendosi di due dita e tenendo gli occhi fissi sulla tastiera. Infine il programma si serve di una sofisticata tecnologia informatica, per creare un ambiente didattico "esperto". Si tratta di un sistema in grado di combinare la valutazione di fatti empirici (come le

violazioni delle regole della dattilografia) con la stessa acutezza ed esperienza di un istruttore umano. Il programma, in pratica, è in grado di adeguarsi ai nostri errori: ad esempio, sapendo che è più facile battere la coppia di lettere "pa" che non la coppia "pl" (che si trova sulla stessa mano),

incontra "paly" al posto di "play" riesce a riconoscere l'errore di battitura, e se l'errore viene ripetuto troppo spesso può decidere di assegnarci una serie di esercizi per allenarci a questo tipo di combinazioni. *Intellitype* è in grado di riconoscere e di proporre esercizi per ben 27 tipi di errori.

Oltre alle sue ottime caratteristiche didattiche, il programma è in grado di fornire dati dettagliati sui nostri progressi, sui nostri errori e sui nostri ritmi, illustrandoli con grafici di vario tipo. In base a questi dati può anche modificare leggermente il corso.

La gestione del programma si effettua tramite il solito mouse e le finestre a scomparsa; attraverso i menu è possibile, prima di iniziare le lezioni vere e proprie, esercitarsi su qualsiasi tipo di combinazione, nel caso non ci si senta in grado di affrontare nuove lezioni o si desideri prolungare il proprio esercizio su qualche punto non particolarmente approfondito.

L'unica (e inevitabile) pecca del programma deriva dalla sua nazionalità. Tutti i testi, infatti, sono in inglese, e, al di là delle difficoltà che avrebbe chiunque, questo rende meno utile il corso a una dattilografa che debba utilizzare la tastiera italiana (leggermente diversa da quella anglosassone). Nonostante questo, chiunque può trarre parecchi benefici dal corso: basterà abituarsi a "trovare" con le dita i tasti con le lettere accentate e cercare di memorizzare le differenze. Create le frasi, in fondo, modificare qualche regola non è poi così difficile.

Da ogni punto di vista, è uno dei migliori programmi del suo genere. Si tratta di un prodotto ben realizzato, affidabile e soprattutto efficace, si potrebbe quasi considerarlo un piccolo esempio applicativo degli studi sull'intelligenza artificiale...

N.F.R

SKYFOX II THE CYGNUS CONFLICT

Computer: C-64/128
Supporto: Cassetta/Disco
Prezzo: L. 22.000/29.000
Produzione: Dynamix
Disponibile presso: Lago (Via Napoleona 16, 22100 Como - 031/300174)

La volpe è tornata... alla velocità della luce! La Electronic Arts ha prodotto la continuazione della saga di *Skyfox*, e c'era da aspettarselo, visto il grande successo ottenuto dalla "prima puntata". In *Skyfox II* cambia però lo

scenario del gioco. Prima, infatti, ci trovavamo di fronte a missioni terrestri di attacco e di disturbo; *Skyfox II* invece trasporta il giocatore nell'affascinante atmosfera della costellazione Cygnus, alla guida, per l'appunto, dello Skyfox II: una potentissima astronave da combattimento capace di prestazioni da fantascienza.

Lo scopo del gioco è quello di portare a termine una delle dieci missioni che si possono selezionare. Le situazioni disponibili sono piuttosto varie, e ogni missione prevede azioni belliche, spionistiche, diplomatiche...

Lo Skyfox è una nave che combatte per la Federazione, e quest'ultima è ancora in lotta con il regno di Xenomorf, al confine del settore controllato dai federati. Proprio in questa zona il giocatore dovrà concentrare i propri attacchi. L'astronave si controlla tramite joystick ed è dotata di armi in grado di distruggere intere basi nemiche.

Anche se la situazione e la trama possono sembrare stimolanti ed emozionanti, il gioco tradisce le sue premesse. *Skyfox II* riprende ben poco dal suo predecessore oltre al nome (evidentemente nato solo per motivi di marketing). Anche se è a tratti divertente scorrazzare per la costellazione (che in realtà è molto estesa) abbattendo tutto ciò che si incontra, la semplicità del gioco e della manovra non possono certo creare grandi emozioni.

La grafica è discreta e riesce a rendere la sensazione di movimento della nave e gli avvicinamenti, si tratta però di immagini piuttosto semplici che, pur utilizzando molti sprite contemporaneamente, non si distinguono certo per accuratezza e realismo.

Proprio a causa della grande quantità di sprite impiegati, la EA definisce *Skyfox II* rivoluzionario. Sembra infatti che la Dynamics, la casa produttrice che ha realizzato questo programma per la EA, abbia inventato un sistema chiamato "Simulated Copper List" che permette al C-64 di gestire contemporaneamente un grandissimo numero di sprite. I risultati, in realtà, si vedono: gli oggetti in movimento che compaiono sul video sono mediamente molto più numerosi del solito. Però nelle fasi più concitate, quando missili, colpi e astronavi si affollano, alcuni sprite scompaiono o lasciano il loro posto a quadrati neri denunciando la scarsa affidabilità di questo sistema.

In definitiva *Skyfox II* è un prodotto discreto: un po' di azione, un po' di strategia e una buona grafica. Ma non si dimostra certo un prodotto rivoluzionario: del vecchio *Skyfox* resta solo il nome.

N.F.R

CRACK

Computer: Amiga
Supporto: Disco
Prezzo: L. 35.000
Produzione: Linel
Disponibile presso: Lago (Via Napoleona 16, 22100 Como - 031/300174)

Ecco l'ennesimo rifacimento di uno dei padri dei videogiochi: *Block buster*. Il famoso gioco della pallina che abbatte muri di mattoni è stato sicuramente uno dei precursori del videogioco moderno, insieme a *Space invaders* e *Pac-man*. *Crack*, ultima creazione della Linel, una software house svizzera, costituisce l'ennesima versione di questo fortunato tema. Dicendo "ennesima", però, ho già messo in evidenza il principale limite di questo programma. Il numero di variazioni sul tema di questo gioco tende veramente all'infinito, e diventa difficile, se non impossibile, apportare modifiche o cambiamenti di effettiva rilevanza. Questo è un tipo di gioco che ha già dato tutto quello che poteva dare.

Crack, in realtà, tranne che per la veste grafica, non è diverso dall'ultima versione di *Block buster* prodotta per bar e sale giochi: *Arkanoid*. Si tratta del tradizionale gioco della pallina e della racchetta, lo scopo del quale è quello di abbattere uno o più muri di mattoni, evitando nel contempo di far cadere la pallina nella parte bassa dello schermo. La racchetta si controlla spostando il mouse a destra e a sinistra e in realtà la precisione di questo modo di controllo fa un po' rimpiangere le rotelle o i paddle delle versioni più tradizionali (sarebbe del resto un po' avventato realizzare un paddle per l'Amiga in funzione di questo solo gioco!).

Come nella versione più evoluta, alcuni dei mattoni contengono speciali bonus che forniscono facilitazioni come una racchetta più lunga, la possibilità di sparare contro i mattoni, la contemporanea presenza di due palline... Dalla parte alta dello schermo, inoltre, compaiono a volte alcuni oggetti (lance, lettere...) che, muovendosi sul campo di gioco, possono disturbare l'azione del giocatore deviando la pallina o, se colpiscono la racchetta, annullare gli effetti dei bonus. Si possono trovare anche mattoni che devono essere colpiti più di una volta per essere abbattuti e perfino mattoni indistruttibili, posti come elemento di disturbo. Chiunque abbia già giocato ad *Arkanoid* avrà sicuramente familiarità con tutti questi "effetti speciali".

Le uniche vere novità del gioco sono la veste grafica e il "coconut game". La veste grafica non modifica per nulla il gioco in sé, si tratta solo di un inutile sfoggio delle capacità dell'Amiga. Il campo di gioco ha uno sfondo composto da immagini di animali preistorici, che si scoprono a mano a mano che i mattoni vengono abbattuti. Le immagini in sé sono indubbiamente ben realizzate ma non hanno alcun effetto se non quello di peggiorare la visibilità dello schermo. Molto simpatica è invece la presentazione del gioco: un'ottima animazione nella quale alcuni uomini delle caverne scolpiscono nella roccia il nome del programma.

L'altra novità del gioco, è costituita, come abbiamo già anticipato, dal "coconut game". Si tratta in realtà di un intermezzo che si presenta ogni volta che si superano tre schermi. Nel coconut game il giocatore controlla con il mouse uno dei cavernicoli visti nella presentazione. L'ometto tiene in mano un cestello, e muovendosi a destra e a sinistra deve raccogliere le noci di cocco che cadono dall'alto dello schermo. Ogni noce di cocco fornisce un punteggio che va ad aggiungersi a quello della partita.

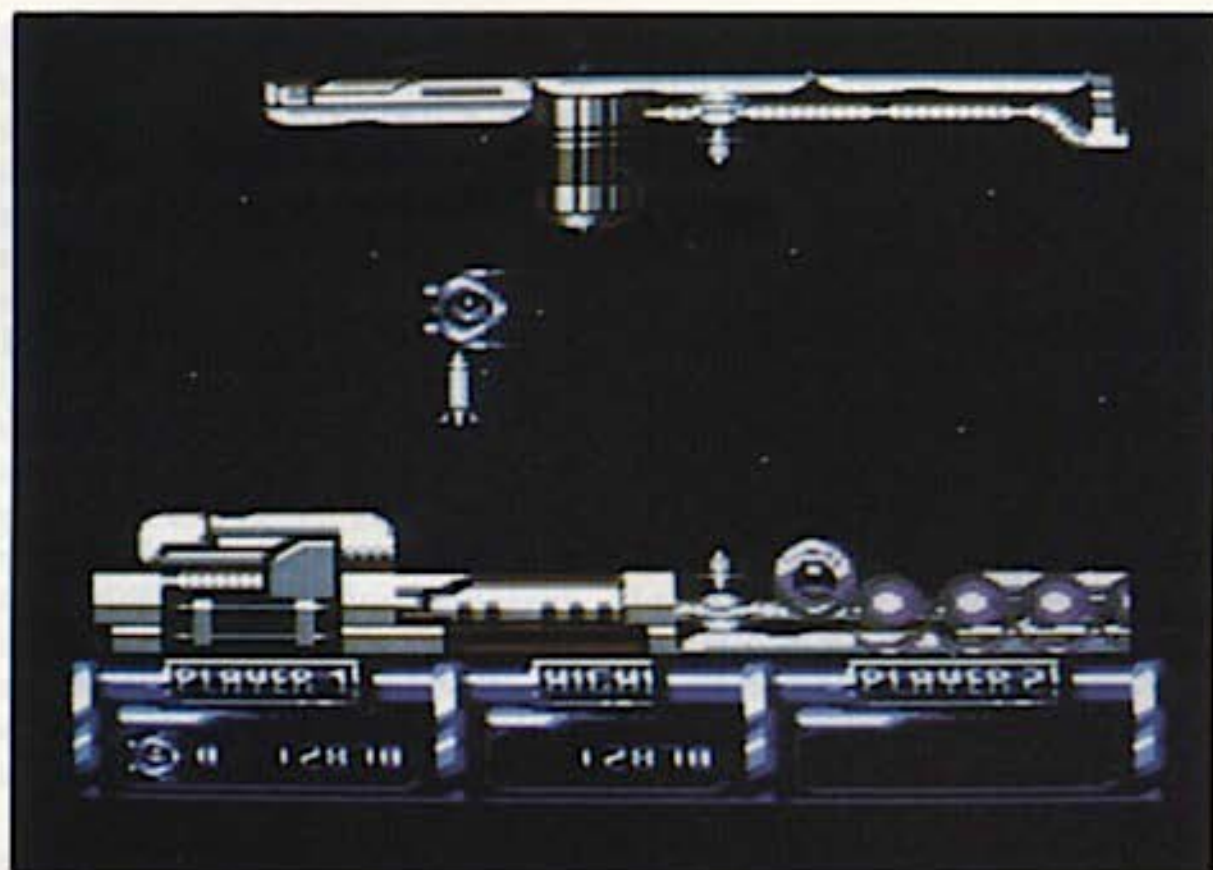
In pratica *Crack* non presenta grosse novità rispetto ai precedenti tentativi di rendere ancora attuale questo gioco. Il programma è indubbiamente ben realizzato specie dal punto di vista grafico, ma non merita certo tante attenzioni: con o senza uomini primitivi, dinosauri e noci di cocco, *Block buster* (o *Break through* che dir si voglia) è sempre lo stesso: una pallina, una racchetta e molti mattoncini... N.F.B.

IO

Computer: C-64/128
Supporto: Cassetta/Disco
Prezzo: L. 18.000/25.000
Produzione: Firebird
Disponibile presso: Lago (Via Napoleona 16, 22100 Como - 031/300174)

Finalmente un gioco d'azione nel senso più puro del termine! Era da parecchio che l'industria del software non produceva uno shoot'em up degno di questo nome, che riuscisse a superare gli schemi tradizionali del genere.

IO, una delle ultime novità nel listino della Firebird, è proprio il gioco che gli appassionati dello spara-e-fuggi aspettavano: semplice, veloce, ricco di fantasia e capace di esaltare l'abilità e la prontezza di riflessi del giocatore.

IO trae la sua ispirazione dai videogame da bar, in particolare da *R.TYPE*, un arcade che sta ottenendo un discreto successo nelle sale giochi italiane. Si tratta di un gioco con schermo a scorrimento orizzontale, nel quale il giocatore manovra una piccola astronave che si può muovere in tutte e quattro le direzioni. Durante la partita si devono affrontare "nemici" dalle caratteristiche diversissime, sia per la forma sia per le tecniche di attacco. Vi sono cannoncini appostati sulla parete superiore e su quella inferiore dello schermo, vari tipi di astronavi che compaiono da destra (l'astronave si sposta sempre da sinistra a destra), e altre (le più pericolose) che possono sbucare da ogni parte. Durante il percorso e alla fine di ogni fase, inoltre, si possono incontrare veri e propri mostri che prima di disintegrarsi devono essere colpiti più volte. Dopo aver abbattuto il mostro che si trova alla fine dello schermo si passa al successivo, dove si scopriranno nuovi nemici e un nuovo scenario.

Il gioco, lo avrete già capito, è piuttosto difficile. È probabile che un giocatore inesperto non riesca a superare neanche i primi 30 secondi, al primo tentativo. Non si tratta però di difficoltà inutili o noiose come spesso capita in videogame di questo tipo; dopo aver preso un po' la mano con i comandi, l'astronave diventa molto maneggevole e le difficoltà possono essere superate grazie alla propria abilità e prontezza di riflessi. Esiste, inoltre, la possibilità di rendere più potente e sicura la propria astronave, catturando i "bonus" sparsi lungo il percorso; si tratta di piccoli rettangoli verdi che se vengono colpiti quattro volte e catturati passandoci sopra, aumentano la potenza di fuoco della navetta. Inoltre, il quarto che catturiamo si trasforma in una piccola sfera che si aggancia alla navetta e permette che venga colpita una volta senza disintegrarsi. I bonus possono anche essere sfruttati "una tantum", cat-

turandoli prima di averli colpiti quattro volte: in questo caso hanno la funzione di disintegrare ogni nemico che si trova in quel momento sullo schermo. Si tratta comunque di aiuti indispensabili per superare alcuni passaggi del gioco, specie quelli che si trovano alla fine di ogni schermo.

Durante le prime partite è molto difficile capire in quale posizione bisogna tenere la propria astronave a seconda dei nemici che si incontrano, anche se gli schermi ripetono più o meno le stesse sequenze di ostacoli: un attimo di ritardo nel movimento della navetta determina quasi automaticamente la perdita di una vita. In pratica, più che colpire indistintamente tutto ciò che capita, è fondamentale evitare di essere colpiti, aggirando i proiettili più lenti, tenendo a distanza le astronavi più insidiose e prevedendo sempre una possibile via di scampo nelle quattro direzioni. Anche la strategia ha una parte decisiva nel gioco e questo sottolinea ancora la cura con cui è stato realizzato.

IO, insomma, è sicuramente uno dei migliori giochi per C-64 che si siano visti negli ultimi tempi. Semplice, veloce e appassionante, interrompe finalmente la più recente tendenza dei produttori di software per C-64, tesi a proporre giochi tattico-strategici di lunga durata e con un'azione ridotta a piccoli e sporadici movimenti del joystick. *IO*, invece, è uno shoot'em up degno di questo nome, con una veste grafica eccezionale per il C-64 e dotato di una giocabilità in grado di fare impazzire anche il più esperto appassionato di spara-e-fuggi.

Se decidete di acquistarlo, mettetelo a confronto con *R.TYPE*, la versione da bar del gioco, tenendo presente l'enorme differenza di potenza dei chip contenuti nelle due macchine. Sarà una bella sorpresa!

N.F.R.

ADVANCED TACTICAL FIGHTER

Computer: C-64/128
Supporto: Cassetta/Disco
Prezzo: L. 18.000/25.000
Produzione: Digital Integration
Disponibile presso: Lago (Via Napoleona 16, 22100 Como - 031/300174)

ATF è un gioco di azione e strategia prodotto dalla Digital Integration che vede il giocatore impegnato in missioni di combattimento su di un modernissimo caccia dal quale il programma prende il nome. Il compito del giocatore è quello di

mantenere il delicato equilibrio tra le proprie forze e quelle nemiche, andando a colpire, con opportune incursioni, i settori nei quali il nemico sta per prevalere. A questo scopo all'inizio del gioco e di ogni missione viene generato un "rapporto di guerra" nel quale vengono messe a confronto le forze del nemico e degli alleati nei vari settori (basi, forze di terra, forze navali, comunicazioni e industrie). In base a questo rapporto il giocatore potrà scegliere quali obiettivi colpire nella successiva missione.

Anche se fino a questo punto il gioco può sembrare interessante e ben strutturato, le delusioni arrivano proprio nel momento in cui si passa all'azione. La "pagina" che segue il rapporto di guerra permette di scegliere la composizione dell'armamento dell'aereo in relazione al peso trasportabile, ma le scelte da fare sono ben poche: il tipo di missile per gli obiettivi a terra (ce ne sono solo due), la quantità di carburante e la quantità di proiettili per la mitragliatrice, destinata esclusivamente al combattimento con gli aerei nemici.

La manovra dell'aereo in volo è fin troppo essenziale, essendo possibile effettuare solo quattro movimenti (alto, basso, destra e sinistra) utilizzando il joystick. Questa semplicità rende l'azione monotona e ripetitiva soprattutto per quanto riguarda i combattimenti, che fanno pensare agli shoot'em up della vecchia generazione. Anche dal punto di vista strategico *ATF* non richiede grande dispendio di energia cerebrale e non lascia spazio né all'abilità né alla fantasia del giocatore: per la vittoria è sufficiente scegliere come obiettivo unità dei settori nei quali il nemico è più forte. Le unità nemiche vengono inoltre inserite in un database che ci segnala la rotta per raggiungerle e la loro distanza, rendendo anche questo compito meno incerto e poco divertente.

ATF, insomma, è un gioco che sulla carta sembra contenere le premesse per realizzare un'ottima simulazione bellica, ma che in alla prova dei fatti dà ben poche soddisfazioni.

N.F.R.

TETRIS

Computer: C-64/Amiga
Supporto: Cassetta/Disco
Prezzo: L. 18.000/25.000/39.000
Produzione: Andromeda Software
Disponibile presso: Lago (Via Napoleona 16, 22100 Como - 031/300174)

Se pensate che il mercato dei videogiochi sia ormai saturo e presenti solo prodotti ripetitivi, poco originali o troppo complicati, allora vi consiglio di mettere da parte per una volta tutte le delusioni precedenti e di provare a giocare con *Tetris*.

La prima definizione che viene in mente è quella di "classico", ma non è così, perché, pur avendo profonde radici nella vecchia Russia, da noi non ha davvero precedenti, se non consideriamo una certa parentela con alcuni test d'intelligenza. Curiosi di sapere di che cosa si tratta?

Inizialmente può sembrare un gioco insulso, un banale puzzle, ma in realtà fin dalle prime partite si viene presi da una vera e propria ossessione, e il paragone più ovvio, in termini di effetto sul pubblico, è il cubo di Rubik. Ma a questo punto una descrizione è proprio indispensabile: alcuni pezzi colorati, di sette diverse forme tutte rigorosamente geometriche, "cadono" dall'alto dello schermo, uno alla volta, adagiandosi sulla linea di demarcazione inferiore. Durante la caduta possono essere fatti ruotare di 90 gradi alla volta in direzione antioraria, semplicemente premendo una o più volte il tasto fire. Scopo del gioco è quello di incastrarli tra loro in modo che formino delle linee orizzontali complete. Ogni volta che si completa una linea, questa scompare dallo schermo, e altri blocchi in bilico cadono verso il basso, per fermarsi su quelli sottostanti. L'obiettivo è quindi la creazione del maggior numero possibile di linee prima che, a furia di accrescere il numero di quelle incomplete presenti sullo schermo, i blocchi in discesa non abbiano più la possibilità di cadere. Il punteggio, proporzionale al numero di linee completate, viene accresciuto anche accelerando la caduta dei pezzi verso il basso (joystick tirato indietro), ma il più delle volte non si riesce a individuare intuitivamente l'incastro perfetto. Un bravo giocatore deve quindi manipolare i vari pezzi facendoli ruotare fino a poterli incastrare con quelli "adagiati" sul fondo; è necessario quindi un ottimo colpo d'occhio e una certa abilità con il joystick.

Da questa descrizione *Tetris* potrebbe sembrare al tempo stesso troppo complicato e troppo elementare per attirare un interesse più che superficiale. Ma non si è forse avuta la stessa impressione all'uscita sul mercato del già citato "cubo"? A questo punto ci pare doveroso differenziare la versione per il C-64 da quella per l'Amiga (a prescindere dalle ovvie differenze nella grafica e nel sonoro): nella prima viene presentato, nella parte destra dello schermo, il pezzo che seguirà quello attualmente in caduta (rappresenta un ulteriore aiuto), mentre la seconda offre in più un diagramma statistico riguardante i pezzi caduti.

Era da tempo ormai che qualcosa di assolutamente nuovo non sconvolgeva il mercato con la sua semplicità; personalmente adoro *Tetris*, forse per la rabbia che riesce a destare partita dopo partita... non passa volta in cui non venga voglia di dire «Accidenti, si incastrava proprio lì!».

È positivo anche il fatto che un errore non condizioni tutta la partita, perché eliminando una linea superiore si crea spesso la possibilità di completare quella sottostante, che il più delle volte si trova in una situazione di stallo. Nel gioco, dunque, il caso è un fattore importante: per esempio, l'uscita di pezzi a forma di cubo è sempre preferibile a quella di pezzi a forma di "S", e spesso una partita finisce nell'ansiosa e vana attesa di un pezzo indispensabile.

Il gioco comprende dieci livelli di difficoltà, e lascia quindi il tempo di entrare a poco a poco nel geometrico mondo di *Tetris*, senza essere traumatizzati da un nubifragio di cubetti. Un altro elemento da segnalare è la colonna sonora che dura ben 25 minuti senza ripetersi ed è molto curata dal punto di vista della composizione e della realizzazione.

Sono tutti piccoli particolari, a ben guardare, semplici idee brillanti ma che non fanno pensare a risultati sconvolgenti; eppure succede spesso che (salvo le novità veramente straordinarie dal punto di vista tecnico) proprio le piccole idee brillanti riescono a creare la differenza tra un programma originale e il solito vecchio gioco. Un voto a *Tetris*? Dieci!

Dimenticavo, se qualcuno di voi sa che cosa significa il nome "Tetris", in russo, ha anche la possibilità di vincere un viaggio in Unione Sovietica. Ma perché tanta presenza dell'Est europeo, in questo gioco? Perché *Tetris* è stato sviluppato nell'Università di Mosca su un IBM, e la sua realizzazione è stata curata dall'Andromeda Software di Budapest, e rappresenta quindi uno dei rari esempi di produzione informatica dell'Est: oggi il software viene anche dal freddo! **L.M.**

TROLL

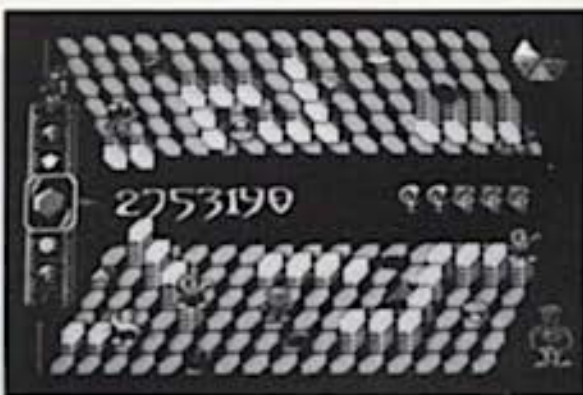
Computer: C-64/128
Supporto: Cassetta/Disco
Prezzo: L. 18.000/25.000
Produzione: Outlaw
Distribuzione: Lago (Via Napoleona 16, 22100 Como - 031/300174)

Troll è uno dei classici giochi in cui la fantasiosa e ricercata storia d'ambientazione è del tutto superflua. Se non ci fosse o se fosse completamente diversa non cambierebbe nulla: il gioco è infatti indipendente dal suo elaborato sfondo e strutturalmente molto semplice. Si tratta di uno schema già visto e rivisto: il protagonista si muove su di un piano, fugge dagli inseguitori e nel frattempo raccoglie qualche oggetto. Il tutto è abbellito da alcuni tocchi di fantasia, come buchi attraverso cui cambiare schermo (cambia ben poco a parte i colori), trappole mortali o piccole piramidi che lanciano il protagonista sul soffitto. Ecco la grande trovata del gioco: tutto si svolge esattamente allo stesso modo, ma in maniera simmetrica... a testa in giù.

Veniamo comunque, per dovere di cronaca, all'ambientazione. Il protagonista si chiama Humgruffin ed è un non meglio specificato Troll, caduto per caso nell'infernale mondo di Narc, terra di caverne di cristallo popolata dai malevoli Goblin. Se l'eroico Troll viene toccato da questi esseri stregati sarà tramutato in pietra. Cosa fare, dunque?

Unica via di scampo la fuga e, ovviamente, per fuggire si devono affrontare complicate procedure... e intanto ogni porta potrebbe essere l'ultima. Bisogna raccogliere le chiavi di cristallo che si trovano in ogni stanza, sconfiggendo così un non meglio precisato incantesimo che si frappone tra noi e la libertà.

Nei movimenti, Humgruffin è impedito da pareti di mattoni costruite dai Goblin e deve schivare i pericolosi buchi della morte. Il gioco si svolge in un ambiente costituito da figure geometriche che vorrebbero, come accade in decine di altri programmi, dare un'illusione di tridimensionalità.

Su questo terreno il Troll si muove, salta e sfugge ai nemici, sfruttando una certa rapidità e un'accettabile precisione, anche se la grafica lascia piuttosto a desiderare. Elementi negativi sono la monotonia degli ambienti, la pessima sezione sonora, la ripetitività del gioco e la mancanza di ogni tipo di sorpresa. Unico arricchimento a questo "attraente" panorama è una ruota della fortuna, sulla sinistra dello schermo, che con il suo girare crea eventi quali la formazione dei buchi per accedere ad altre stanze o per eliminare gli inseguitori. Per azionare la ruota della fortuna si deve saltare "freneticamente" (per usare le parole del manuale, tradotto in italiano a cura della Lago) sul rospo che qualche volta spunta dal pavimento a mattonelle, convocando così lo gnomo Fjavar che a quanto pare è l'unico autorizzato a metter mano alla magica ruota.

Più fantasia nei propositi che nella pratica, dunque. Un gioco come tanti, che lascia un senso di *deja vu*, e non induce certo a lasciarsi catturare dall'infernale mondo di Narc.

A.M.

SOFTWARE HELPLINE

John Brenner, Boston Celtic

Non è il caso di soffermarci in una presentazione di questa "adventure", già ben nota ai lettori di *Commodore Gazette* (vedere il numero di marzo/aprile).
La soluzione, realizzata da Fabio Carignano, consiste in una descrizione generale dell'azione seguita dall'elenco dei "passi" da compiere.

Prima parte: hai sete, quindi entri nel drugstore ed esamini gli scaffali alla ricerca di una bottiglia di whisky. All'arrivo dei rapinatori ti butti faccia a terra, sollevi gli occhi ed esamini le scarpe del bandito. Abbassi la testa e quando i rapinatori se ne vanno ti alzi e li insegui. Apri la porta, impugni la pistola e... peccato, ti sono sfuggiti. Osservi la strada, raccogli la maschera e torni a casa per riposarti.

(entro nel drugstore, esamino gli scaffali, mi sdraio faccia a terra, sollevo gli occhi, esamino le scarpe verdi, abbasso la testa, mi alzo in piedi, inseguo i rapinatori, spingo la porta, prendo la pistola, esamino la strada, raccolgo la maschera).

Seconda parte: sei arrabbiato, anzi arrabbiatissimo, ma ti viene in mente il nome di un amico che può aiutarti: Stan Gilbert. Ti rechi da lui, entri al *Daily News* e raggiungi Stan passando dalle scale. Lo saluti, lo segui e lui ti consiglia di andare dal suo amico Parker, oppure in archivio. Vai all'archivio, apri la porta, entri ed esamini la raccolta. Leggi l'articolo, cerchi l'indirizzo ed esci. Vai nell'ufficio di Parker, lo saluti e ti presenti. Avute le informazioni lo ringrazi ed esci dall'edificio.

(arrabbiato, arrabbiatissimo, Stan Gilbert, vado da Stan, entro, vado a destra, saluto Stan, lo seguo, sì, ciao Stan, vado in archivio, apro la porta, entro, esamino la raccolta, leggo la notizia, cerco l'indirizzo, esco, vado da Parker, saluto Parker, John Brenner investigatore privato, sì, continua Parker, gli sorrido, esco dal Daily News).

Terza parte: vai al magazzino teatrale e suoni il campanello. Ti presenti e mostri la maschera come prova. Ringrazi e vai al Play Off. Entri, vai al banco e ordini

birra. Scopri che JJ significa Jordan Johnson, chiedi dove puoi trovarlo. Vai in palestra, esamini i giocatori e prendi la palla. Chiedi dov'è JJ e lo raggiungi negli spogliatoi. Apri la porta, lo insegui, entri nel corridoio. Ti alzi e catturi l'uomo che ti ha fatto cadere. Chiedi dov'è andato JJ e telefoni alla polizia.

(vado al magazzino teatrale, 112 85esima ovest, suono il campanello, John Brenner investigatore privato, fare delle domande, prendo la maschera, mostro la maschera, continui Tanner, ringrazio Tanner, vado al Play Off, entro, vado al banco, ordino una birra, sì, sono tifoso del Boston Celtics, JJ, dove posso trovarlo, vado alla palestra, entro, esamino i giocatori, la prendo, interrogo il giocatore, Jordan Johnson, vado negli spogliatoi, apro la porta, lo inseguo, entro nel corridoio, mi alzo, catturo l'uomo, dov'è andato, telefono alla polizia).

Quarta parte: vai al molo 20, cerchi il dock abbandonato e prendi la pistola. Cerchi di raggiungere JJ (vai sempre a est e a sud finché non lo raggiungi). Gli dici di gettare la pistola.

(al molo 26, cerco il dock abbandonato, prendo la pistola, entro, sequenza di est e sud..., getta la pistola).

Beyond Zork

In *Beyond Zork* ci sono molti elementi casuali: le locazioni in alcune aree e gli oggetti che vi si trovano cambiano di volta in volta, e gli incantesimi (spell) spesso hanno nomi diversi, o devono essere lanciati con oggetti (item) diversi. Per queste ragioni non viene fornita una mappa, che non servirebbe a nulla se non foste tanto fortunati da incappare nell'identica nostra situazione. La soluzione quindi dev'essere fornita in una forma piuttosto inconsueta.
Anziché dare un percorso passo-per-passo, questa volta diremo che cosa fare e quali incantesimi sono necessari in ogni particolare zona. Gli effetti di ogni oggetto magico variano da partita a partita, così le frasi riportate si riferiscono semplicemente all'incantesimo: Thing of Annihilation, per esempio. Si può chiedere alla donna che si trova nel Magick Shoppe di identificare incantesimi e relativi oggetti magici. Se non possedete uno degli oggetti qui menzionati, provate a guardarvi attorno... non sarà difficile trovarlo. Abbiamo messo in evidenza le collocazioni di quelli che non si notano immediatamente.

Hilltop, Sea e Tavern
Andate a nordovest fin dall'inizio e prendete l'erba magica (weed) a *Edge of Storms*. Al porto esaminate le vele (sailor's canvas) e prendete (get) un pezzo di legno galleggiante (driftwood). Usatelo come arma. Prendete la lanterna (lantern) fuori dalla taverna (Tavern), entrate e aspettate di udire la parola *helmet* da un brigante. Tentate di andare a ovest, quindi prendete il pugnale (dagger).

Wine Cellar e Kitchen
Quando entrate per la prima volta nella cantina (cellar) prendete il rotolo (scroll) e leggete (se la luce se ne va, dite la parola). Schiacciate (squeeze) il muschio (moss) sul muro per avere *Dexterity* per salire sul *Bottom Stack* e prendere la bottiglia di vino (wine bottle). Eliminate (slay) lo scheletro (skeleton), prendete l'amuleto (amulet) e mettetevelo addosso (wear). Prendete la corona (crown) e cercate (search) il *nest* per la moneta (coin). Vendeteli (sell) entrambi. Dopo che la porta si è chiusa (slammed) dietro di voi, *read the amulet through the bottle* e dite la parola per prendere uno *Strenght boost* con il quale potrete rompere la porta. Conservate la lanterna spegnendola (shutting off) il più presto possibile. Date la bottiglia al cuoco (cook). Andate a est e prendete la coperta (rug), buttando (pushing) la cipolla (onion) sul *cliff wall*.

Cliff e Lighthouse
La risposta all'indovinello è la parola *lightning*, che vi permette di entrare nella *lighthouse*. All'interno avrete bisogno di *salt* (dai *tidal flats*), *rug*, *Thing of annihilation*, *dagger e onion*. Colpite (bash) il ragno (spider) e buttate il sale (throw salt) sulla lumaca (slug). Strofinate la coperta (rub the rug) e toccate (touch) il *Dust Bunny*. Prendete l'anello (ring) e infilatelo (wear). Vendete (sell) la *card* del *bubblegum*. Tagliate (cut) la cipolla (onion) con il pugnale (dagger) per stordire il *Dornbeast*, puntate la *Thing of annihilation* su di lui e prendete la cesta (chest) impadronendovi di tutto (get all). Cercate le macerie (debris) nella stanza superiore (top room) e vendete il sestante (sextant).

Accardi by the Sea
Date la cesta (chest) al *Monkey Grinder* quando vedete che ha schiacciato la *warning nymph*. Prendete *palimpsest* e *vague outline* dalla cesta. Prendete il *gurdy*. Ora potete vendere la cesta. Comprate l'ascia (axe), impugnatela (wield) e usate su di essa lo *Scroll of Honing*. Usate l'ascia per eliminare (slay) il *puppet*, quindi puntate la *Thing of annihilation* verso *Hellhound*. Prendete (get) tutto quello che trovate nella foresta. Mangiate (eat) la *fishcake* per avere un aumento d'intelligenza.

Plane of Atrii
Dite la parola dalla *gating spell* (dalla cesta) per entrare nella piana di Atrii. Muovetevi in giro finché non trovate un posto dove la fabbrica di tempo spreme *shut* per produrre un *vague outline*. Il *vague outline* che si trovava nella cesta diventa una *Phase Blade* quando entrate nella piana di Atrii; usatela per tagliare (cut) il *vague outline* che vi blocca la strada. Nell'area bloccata troverete gli *Implementors*. Aspettate finché non lasciano cadere il *coconut* e vi danno il *goost*. Molto probabilmente vi teletrasporteranno al *billboard*.

Fields of Frotzen
Il *goost* vi proteggerà dal temporale. Prendete il *four-leaf clover*, quando lo trovate. Attendete una farfalla (butterfly) per atterrare sul *goost*. Aprite il *gurdy* e metteteci dentro il *goost*. Chiudete il *gurdy, set the dial to clock* e girate la manovella (cranck) a sinistra (left). Aprite il *gurdy* e prendete il *caterpillar* prima che scappi. Togliete il *goost* e chiudete il *gurdy*. Trovate lo spaventapasseri (scarecrow) circondato di grano. Girate il *dial* sul *gurdy* per sbirciare, quindi girate la manovella (cranck) a destra (right). Fate attenzione ai cenci (rags) dello spaventapasseri (scarecrow): i *Corbies* hanno paura di quel colore. Quando la casa (farmhouse) è crollata, entrate e restateci finché la tempesta non si calma.

Land of Froon
Uscite dalla casa. Esaminate i fiori (flowers). Quando vi viene offerta, prendete la chiave (key) che ha lo stesso colore degli stracci dello spaventapasseri. Quando verrete trasportati indietro attraverso i campi potrete superare i *Corbies* e prendere la *Compass Rose*.

Jungle
Uccidete oppure puntate la *Thing of annihilation* verso il coccodrillo (crocodile) e *bloodworm*. Trovate l'idolo (Idol), quindi l'*Hungus*. Attaccate (attack) il bambino (baby). La madre vi attaccherà e vi seguirà fino all'idolo. Arrampicatevi sull'idolo (climb the Idol). Finché la madre è ancora giù prendete il gioiello (jewel). All'interno dell'idolo accendete la lampada (turn on the lamp). Spremete (squeeze) il muschio (moss), quindi puntate la *Thing of eversion* contro il muro (wall). Trovate la madre e puntate verso di lei la *Thing of eversion* per avere il gioiello. Poi puntate la *Thing of levitation* sul bambino (baby). Potete vendere la zanna (tusk).

Thriff
Guardate sotto il banco (pew) nella cappella (chapel) e cercate la fiala (vial). Ascoltate il Cardinale (cardinal). Esaminate il glifo (glyph), quindi andate a ovest della radura (clearing). Il *caterpillar* spaventerà i *Trees*. Vendete l'ornamento (ornament). Aprite la cassetta della posta ed esaminate

ne il contenuto (content) completamente. Questo vi procurerà un bulino (burin). Andate a sud e cercate il *bench* per *get* il *Black Hemisphere*. Andate a nord e a ovest per uccidere lo *Snow Wight*. Forse dovrete recuperare le forze più di una volta per riuscirci, oppure usare la *Thing of annihilation*. Vendete lo *snowflake*. Andate a est e puntate la *Thing of dispell* contro la cupola (dome). Andate a *Thriff* prima di *get fried*. Siate sicuri di avere addosso (wear) l'anello (ring), quindi tornate alla radura. Con il bulino (burin), disegnate il glifo (glyph) nella lava bollente. Chiedete il *reliquary* al Cardinale e da lì *get* il *White Hemisphere*. Andate a nordovest e nascondete (hide) le impronte della civetta (minx). Quando il cacciatore (hunter) se n'è andato, coccolate (pet) la civetta (minx). Aspettate (wait) finché la civetta non scava fuori un tartufo (truffle) e lasciate che lo mangi. Non dimenticate il muro soleggiato (sunlit wall) a ovest. Ricordate, dovete raccogliere (pick up) la civetta (minx) e portarla con voi.

Stable
Prendete il ferro di cavallo (horseshoe). Per prima cosa baciate (kiss) il corno dell'unicorno (unicorn's horn) perché porta fortuna, poi dite la parola (word) sull'amuleto (amulet) e colpite (smash) la *stable door*. Oppure potete usare la *Thing of levitation* per liberare l'unicorno e prendere la sella (saddle).

Pterodactyl
Assicuratevi di avere con voi l'erba magica (weed) e la *Thing of anesthesia*, e di aver abbastanza spazio per tenere la freccia (arrow) e lo zufolo (whistle). Puntate la *Thing of anesthesia* contro lo pterodattilo (pterodactyl). Prendete la freccia, quindi mettete l'erba (weed) sulla ferita. Prendete lo zufolo (whistle) e mettetevelo addosso (wear). Mettete la sella (saddle) allo pterodattilo (pterodactyl). Prendete la civetta (minx) e salite (board) sullo pterodattilo. Per volare si deve andare in su (up) e puntare la *Compass Rose* nella direzione opposta a quella voluta. Volate finché non sarete sopra il castello (castle) vicino a *Thriff*, quindi atterrate (land).

Castle
Nascondetevi dietro *the Morgia bush*. Prendete un po' delle frasche (bush) e mangiatele (eat). Aspettate che il *platypus* arrivi e se ne vada, quindi uscite dal vostro nascondiglio. Aprite (open) la statua (statue) e prendete il *Crystal Jar*. Suonate lo zufolo (blow the whistle), prendete la civetta (minx) e salite (board) sullo pterodattilo. Volate fino al *Magick Shoppe*.

End Game Preparations
Vendete il gioiello (jewel) e comprate la clessidra (hourglass). Andate fino alla quercia (oak tree) che si trova nella zona crepuscolare (twilight part) della foresta (forest). Aspettate finché la civetta (minx) scava fuori un tartufo (truffle). Prendetelo. Andate fino alla radura (clearing) nella foresta. La risposta all'indovinello sulla roccia è *Youth*. Entrate nel *Pool of Radiance*. Il tartufo verrà conservato per tutto il tempo. Chiamate (whistle) lo pterodattilo. Mettete al sicuro il tartufo nel *pack* prima di prendere la civetta (minx), quindi volate alle *Ruins*.

Ruins
Colpite (bash) il Ghoul. Gettate il *vhal* all'*Undead Warrior*. Restate sotto l'arco (arch) nella piazza (plaza) e girate la clessidra (hourglass). Andate doppiamente a sud (twice south), cioè indietro nel tempo. Prendete il tartufo e aspettate (wait) che il cavallo (horse) del principe (prince) cada nel fossato (trench). Buttate (throw) il tartufo (thruffle) nel fossato. Ruotate (turn) di nuovo la clessidra e andate avanti nel tempo (north) finché non raggiungete la desolazione (desolation). Aspettate che la civetta (minx) scavi fuori un tartufo (thruffle). Prendete l'elemetto (helmet) e mettetevelo (wear). Girate la clessidra e tornate alla piazza (plaza). Usate lo pterodattilo, *Scroll of gating* o *secall* per tornare al *Magick Shoppe*. Comprate la pozione di *Enlightenment* se non l'avete già. Scuotetela (shake) e bevetela (drink). Adesso siete abbastanza intelligenti per continuare. Mettete il *peg* nel *Black Hemisphere* nel buco nel *White Hemisphere*. Guardate (look) nella *Gray Sphere* e ricordate la parola magica che troverete.

Underground End Game
Assicuratevi di avere i seguenti oggetti: *rabbit's foot, horseshoe, four-leaf clover, Jar of Mirrors* e *lantern*. Strofinate (rub) la zampa di coniglio per avere un po' di fortuna extra. Andate alla *cliff wall* vicino a *Thriff* (dove avete trovato la civetta). Pronunciate la parola magica della sfera grigia (gray sphere) e aprite la porta. Entrate nella prima stanza sotterranea (underground room). Forse comparirà un *Lucksucker*. Buttategli uno dei vostri oggetti della fortuna ciascuna delle tre volte che apparirà. Nella prima camera sotterranea aprite la giara (jar), prendete il cerchietto (circlet) e digitate "blow bubble". Puntate lo specchio (mirror) in modo tale da illuminare con la luce del sole gli angoli nascosti e il fondo del tunnel (per esempio puntando lo specchio a nord la luce risulterà diretta da nordest a nordovest, o viceversa).

The Treasure Chamber
Tenete le *blowing bubbles* durante l'esplorazione delle stanze finché non trovate la stanza del tesoro (treasure chamber). Dirigete la luce con lo specchio lungo la via più diretta verso la stanza. Gli specchi non durano a lungo, così anche il vostro si romperà (non è necessario "dip circlet"; "blow bubble" sarà sufficiente e farà risparmiare tempo). *Blow* l'ultimo specchio nella stanza del tesoro e dirigete la luce per illuminare l'ombra nell'angolo. Se avete abbastanza *compassion* dovreste riuscire a distruggere l'*Ur-Grue* e potrete quindi cercare il bottino (plunder) e prendere il *coconut*. Se tentate di uscire dalla caverna sarete fermati da un terremoto. Niente paura: ve la caverete e sarete... Beyond Zork!

CURIOSITÀ E INFORMAZIONI UTILI

C-64: se aspettate un paio di minuti dopo averlo caricato, *Raid over Moscow* inizia la partita automaticamente. Premendo il pulsante Fire potrete prendere il controllo del gioco nel momento che preferite.

C-64: con *Aliens* potete passare allo schermo successivo premendo contemporaneamente i pulsanti @, *, P e RESTORE.

Amiga: a proposito del programma di volo *Flight Simulator II*, c'è qualcosa che forse non avete mai provato a fare. In WWI decollate, attraversate il fiume e atterrate. Fate una dichiarazione di guerra (Shift W) e aspettate che ci siano sei aeroplani nemici dietro di voi. Potete facilmente abbatterli uno alla volta. Vi attaccherebbero a qualunque altitudine, ed è molto più facile sparare loro da terra, senza preoccuparsi di dover controllare il volo. Quando avrete eliminato tutti i nemici, decollate, dirigetevi verso i depositi di bombe e bombardateli tranquillamente senza essere molestati da continui attacchi. Quindi tornate alla base, fate rifornimento e decollate di nuovo. Questa volta i nemici saranno pronti... ma con un po' di pratica potrete diventare abbastanza veloci da riuscire ad attraversare il fiume, atterrare e ripetere la stessa procedura.

C-64: Premendo F5 in *Championship Baseball* per mettere il gioco in pausa, potrete attraversare tranquillamente tutte le basi dei vari giocatori. Assicuratevi di avere almeno un uomo in prima base e premete il tasto di pausa non appena il pitcher si prepara al lancio.

INPUT/OUTPUT

INPUT/OUTPUT SVELA I SEGRETI DEL VOSTRO COMPUTER

C-64/128 in modo 64

***162 La classifica computerizzata -** Questo semplice programma realizza sullo schermo un comodo generatore di classifica. Viene richiesto il punteggio e il nome del giocatore. Quando il programma ha ricevuto i dati visualizza la classifica corrente, composta da quattro posizioni. In seguito l'utente può continuare a immettere punteggi e nomi di giocatori, e il programma provvede ogni volta a visualizzare la classifica aggiornata dei top-4.

```
10 REM ********************
20 REM
30 REM  Generatore di
40 REM   classifiche
50 REM
60 REM ********************
100 GOTO 160
110 FOR K=0 TO 4: IF PU>AP(K) THEN 130
120 NEXT: RETURN
130 INPUT "INIZIALI DEL CONCORRENTE";AN$
140 FOR I=5 TO K-1 STEP -1: AP(I)=AP(I-1):
   AI$(I)=AI$(I-1):NEXT:AI$(K)=AN$:AP(K)=PU
150 PRINT CHR$(147): FOR I=0 TO 4:
   PRINT I+1,AP(I),AI$(I):NEXT:RETURN
160 INPUT "PUNTEGGIO";PU: GOSUB 110: GOTO 160
```

***163 Il fascino dei numeri romani -** Gli algoritmi per trasformare un numero dalla notazione araba a quella romana sono tantissimi. Quello che qui proponiamo ha il pregio d'impiegare un numero ridottissimo di istruzioni, e quindi di essere particolarmente veloce.

```
10 INPUT "NUMERO DA TRASFORMARE";N
20 READ V,R$: IF N=>V THEN N=N-V: PRINT R$;
30 IF N>0 THEN 20
60 DATA 1000,M,1000,M,1000,M,900,CM,500,
   D,400,CD,100,C,100,C,100,C
70 DATA 90,XC,50,L,40,XL,10,X,10,X,10,X,
   9,IX,5,V,4,IV,1,I,1,I,1,I
```

Commodore 128

***164 Clock a 2 MHz per il C-64 -** È noto che il C-128 consente di emulare il C-64, ma non tutti sanno che una volta passati in modo C-64 è ancora possibile

Input/output rivela ogni mese ai suoi lettori routine di programmazione per C-64, C-128 e Amiga. Avete in mente un'idea particolare, una routine, un utile stratagemma di programmazione, avete sviluppato un breve programma o in generale pensate che le Vostre conoscenze possano interessare l'utenza Commodore? Allora scrivete, e inviate gli eventuali listati stampati su carta bianca o salvati su disco a:

COMMODORE GAZETTE
Input/Output
Via Monte Napoleone,9
20121 - Milano

disporre dei 2 MHz, contro il clock da 1 MHz tipico del C-64. Eseguendo l'istruzione POKE 53296,1 si attiva il clock da 2 MHz in modo C-64, mentre con l'istruzione POKE 53296,0 si ritorna al normale clock da 1 MHz.

Pur ottenendo una velocità d'esecuzione delle istruzioni praticamente doppia, lo schermo si riempie di confusi disegni e sembra inutilizzabile. Non sono inoltre possibili gli accessi al disco in quanto le delicate temporizzazioni non corrispondono più a quelle previste dal Kernel.

Per eliminare i fastidiosi disegni sullo sfondo, si può disabilitare temporaneamente lo schermo tramite l'istruzione POKE 53265,0 e riabilitarlo con POKE 53265,27. Ma si può compiere la stessa operazione impiegando ancora il misterioso registro allocato all'indirizzo 53296, tramite l'istruzione POKE 53296,3, che abilita il clock da 2 MHz e disabilita lo schermo. Questo secondo modo per disattivare lo schermo non si limita a disabilitare il raster, e non può essere impiegato in modo C-128 in quanto blocca il computer.

In definitiva, la prassi da seguire per impiegare il clock da 2 MHz in un programma Basic è la seguente:

```
100 REM Programma
...
500 POKE 53296,3: REM Attiva il clock da 2 MHz
    e disabilita lo schermo
510 REM Segmento del programma che non genera output
    sullo schermo
...
900 POKE 53296,0: REM Attiva il clock da 1 MHz e abilita
    lo schermo
```

Carlo Casagrande
Conegliano (TV)

***165 Istruzione DATA semplificata -** Inserire i dati in un programma è sempre un'operazione noiosa e ripetitiva. Poter almeno evitare di riscrivere continuamente l'istruzione DATA è già un sollievo. Ecco una procedura che consente d'inserire il comando DATA con la semplice pressione del tasto F7.

```
Attivare il monitor scrivendo MONITOR e
Return
Digitare: >01030 4E 0D 0E 44 41 54 41 4D e
premere il Return
Premere X per disattivare il monitor
```

Seguendo queste semplici istruzioni, l'inserimento dei DATA diventa molto più agevole. È sufficiente premere F7 perché appaia uno spazio e l'istruzione DATA. Se poi si attiva anche il comando AUTO per la numerazione automatica delle linee, lo sforzo diventa davvero minimo.

Gianfranco
Biella

Commodore Amiga

***166 Controlliamo lo stato della stampante -** Quando si realizza un programma in Amiga Basic può diventare necessario operare un controllo sullo stato della stampante. La semplice routine che presentiamo verifica il corretto funzionamento di una qualsiasi stampante collegata all'Amiga tramite la porta parallela.

```
SUB PChk STATIC
Chk:
LOCATE10,10
IF PEEK(12570624&) AND 1 THEN PRINT "CONTROLLARE LA
STAMPANTE"
IF NOT PEEK(12570624&) AND 1 THEN EXIT SUB
GOTO Chk
END SUB
```

Si dovrebbe eseguire questa routine ogni volta che si accede alla stampante da Amiga Basic. È sufficiente inserire nei propri listati l'istruzione PChk o CALL PChk, e gestire il risultato nel modo che si ritiene più opportuno, magari visualizzando un requester che informi l'utente.

***167 Shortcut per i comandi dell'Amiga -** Molto spesso diventa noioso digitare alcuni comandi dell'AmigaDOS come INFO. Sarebbe molto più comodo poterli indicare con una semplice combinazione di tasti. Il sistema per ottenere questo tipo di shortcut esiste ed è anche molto semplice: consiste nell'assegnare una lettera come un dispositivo. Per esempio, impartendo il comando

```
Assign I: C/INFO
```

si trasforma il comando in un device logico di nome "I:". Ogni volta che si digita I viene ora eseguito il comando INFO. Questo impiego dei dispositivi occupa solo 80 byte, e non altera il normale funzionamento dei programmi.

I PROGRAMMATORI DEL MONDO AMIGA A CONFRONTO

All'annuale convegno dei programmatori nascono gli "Amiga Working Groups". Sotto esame il nuovo sistema operativo. Presentato il nuovo Enhanced Chip Set per l'Amiga

di Matthew Leeds

Ogni anno la Commodore USA organizza un convegno che riunisce l'intera comunità dei programmatori dell'Amiga: si discutono le nuove linee di sviluppo, i nuovi prodotti hardware, le variazioni nel sistema operativo... è il terreno più adatto per uno scambio di idee. Quest'anno il congresso si è tenuto a Washington D.C., la capitale degli Stati Uniti. Una scelta senz'altro ideale per un incontro che si propone di essere una tribuna aperta, e di favorire il libero scambio d'idee e d'informazioni fra i programmatori di tutto il mondo.

In effetti questo è stato davvero un congresso speciale, che si è distinto dai precedenti sotto molti punti di vista: la Commodore, per la prima volta, non ha richiesto che gli intervenuti firmassero un impegno di non-divulgazione di quanto sarebbe emerso nel corso degli incontri. Inoltre il numero dei partecipanti ha superato le previsioni di circa il 50%: sono arrivati programmatori dalla Turchia, dall'Australia, dalla Danimarca... tutti intenzionati a migliorare il loro bagaglio di conoscenze e comunicare agli altri i propri progressi.

Nel corso del congresso la Commodore ha presentato nuovi prodotti hardware, discusso il nuovo sistema operativo, spiegato le appropriate tecniche di programmazione, approfondito i progressi nell'interfacciamento hardware, annunciato i piani futuri per l'Amiga, e infine si è fatta promotrice di seminari sul marketing e ha fornito la sua assistenza ai programmatori meno esperti.

Il progetto ARP

La Commodore ha anche aperto uno spiraglio alla possibilità di garantire un'assistenza semi-ufficiale ai programmatori che cercano di battere nuove strade nello sviluppo di software per l'Amiga: in precedenza, per i programmatori indipendenti non c'era nessun modo (se non indirettamente) d'influire sui nuovi sviluppi dell'Amiga. Un esempio di questa nuova linea è il progetto ARP, un consorzio di programmatori che si è dedicato alla stesura di programmi che dovrebbero rimpiazzare un certo numero di comandi dell'AmigaDOS con comandi più rapidi ed efficienti (al progetto ARP abbiamo dedicato parte dell'articolo "La costante evoluzione del mondo Amiga" pubblicato nel numero di marzo/aprile 1988 di *Commodore Gazette*, n.d.r.). Per molte ragioni la Commodore era stata a lungo restia a dare al progetto ARP un'approvazione ufficiale (cioè utilizzarlo nel disco del *Workbench* ufficiale), mentre ora ha assunto al riguardo una posizione più sfumata.

Nasce l'AWG

Perry Kivolowitz ed Eric Lavitsky, dell'ASDG, hanno avanzato la proposta di creare un'organizzazione chiamata "Amiga Working Groups" (gruppi di lavoro Amiga) che dovrebbe riunire programmatori accomunati dall'interesse nello sviluppo di nuovo software e hardware per l'Amiga. Ogni novità prodotta da un membro dell'AWG rimarrebbe di proprietà dell'organizzazione, e l'AWG agirebbe come agente della comunità dei programmatori.

L'AWG intende garantire ai suoi membri il libero uso di tutto il materiale sviluppato dall'organizzazione. La Commodore ha accettato di partecipare a

associazione e d'inserire due suoi rappresentanti nel comitato direttivo dell'AWG. Ha inoltre acconsentito a servirsi dell'AWG come di un canale formale preferenziale per suggerimenti e comunicazioni per lo sviluppo futuro dell'Amiga.

Quali saranno effettivamente i risultati a lungo termine nessuno può dirlo, ma questa partecipazione segna comunque l'inizio di una politica di "porte aperte" da parte della Commodore che certamente sarà di beneficio per tutti.

Novità hardware per l'Amiga

La Commodore ha fornito i dettagli relativi ai suoi nuovi prodotti hardware. Alcuni erano già stati presentati al Comdex o ad Hannover, ma questa volta la descrizione è stata molto più dettagliata ed è stata data una chiara immagine delle intenzioni della Commodore riguardo alle sue nuove iniziative.

Il monitor monocromatico ad alta risoluzione A2024 sta per diventare parte integrante della linea di prodotti per l'Amiga. La Commodore intende presentarlo come un optional che diventerà indispensabile nei prossimi anni. Gran parte dello speciale software progettato per rendere compatibile l'A2024 sarà alla base della versione 1.4 del sistema operativo dell'Amiga: i progetti per l'AmigaDOS 1.4 prevedono già il pieno supporto in ROM per il monitor A2024.

La scheda A2620 sarà disponibile in due versioni (da 2 o da 4 MB di memoria RAM a 32 bit) e fornirà l'opportuno supporto a un'ampia varietà di coprocessori, come il 68881 o il 68882 a 14, 20 o 25 MHz. Il 68851 incorporato provvederà alla gestione della memoria in pagine virtuali richiesta per lavorare in ambiente UNIX. Molti programmatori si sono spesso trovati a disagio con l'attuale scheda 68020 della CSA per l'A2000, e l'introduzione da parte della Commodore di un'efficiente alternativa aprirà la strada a nuovi sviluppi software che sfruttino le maggiori potenzialità di questo processore.

Dal momento che il *Kickstart 1.3* offre la possibilità di autoboot da hard disk, la Commodore ha trasformato il suo controller A2090 in modo che se ne possa servire. Il nuovo controller è chiamato A2090A e include su ROM i codici necessari per eseguire il boot direttamente dall'hard disk.

Benché non ci sia modo di modificare gli A2090 già in commercio per fornirli di questa capacità, su suggerimento di un progettista di hardware si è discusso sull'eventualità di disegnare una scheda poco costosa che contenga la ROM necessaria per l'autoboot e sia in grado di passare poi il controllo all'A2090. Questo sistema richiederebbe però l'impiego di un ulteriore slot.

Molti programmatori esterni hanno già realizzato per l'A2000 dei controller in grado di eseguire l'autoboot da hard disk. Ve ne sono due in stile hardcard, con un drive da 3,5" montato direttamente sul controller: questo permette di mantenere libero lo spazio per un drive da 5,25" destinato a un eventuale disk drive, o a uno streamer.

Molti programmatori si sono dimostrati preoccupati per l'uso dei controller DMA durante la generazione di animazioni in alta risoluzione ottenute con due buffer. Sembra che il controller DMA richieda una certa quantità di cicli di clock della CPU, e che l'animazione in alta risoluzione con due buffer ne riduca il numero al di sotto di un livello accettabile. Come risultato, si ottengono errori in lettura o in scrittura sull'hard disk. Finora questo problema è stato rilevato solo nell'A2090 della Commodore. Benché la Commodore dichiari di aver completamente riscritto il software per il driver nel tentativo di modificare la situazione, i programmatori con cui ho parlato non sembrano avere grande fiducia nell'efficacia di questa soluzione.

I nuovi chip

È stato discusso nei dettagli il nuovo Enhanced Chip Set (ECS). Si tratta di un set di tre chip che

include un "Fatter Agnus", una "Denise" e un "Gary". Al momento attuale il "Fatter Agnus" è completo, e ne sono stati forniti alcuni campioni ai programmatori. I nuovi chip Denise e Gary saranno forniti in seguito. L'ECS è un upgrade per tutti gli A500 e i 2000 dotati di una scheda madre a due strati (B2000). Per l'installazione è necessario aprire il computer, rimuovere gli attuali chip ed eliminare alcune piste o cambiare dei jumper (o ambedue le cose). I possessori di un 2000 con una scheda madre a 4 strati (A2000) probabilmente dovranno rimpiazzare la scheda madre per usare l'ECS.

Le caratteristiche principali dell'ECS sono una memoria RAM da 1 MB, programmabilità secondo gli standard NTSC e PAL (tramite lo stesso chip), blit più larghi e modo superhires che offre una risoluzione 640 x 400 (o 640 x 512) senza interlace (sfarfallio). Il modo superhires richiederà la versione 1.4 del sistema operativo, benché in effetti ci sia la possibilità di modificare il software speciale usato dall'A2024 per dare supporto al modo superhires prima che la versione 1.4 sia effettivamente disponibile.

È il nuovo Agnus a fornire tutta la memoria, cioè 1 MB di RAM, per l'A500 e per l'A2000. Lo stesso chip darà supporto a 2 MB di RAM per i futuri modelli della linea Amiga, ed è stato discusso a lungo se sia possibile o meno modificare l'hardware delle macchine attualmente in commercio per dare supporto fin d'ora al secondo MB di RAM.

Altre modifiche al Fatter Agnus includono cambiamenti nelle locazioni riservate ai canali audio, nel puntatore al blitter, nella locazione per le istruzioni al coprocessore e nelle locazioni DMA per il drive, che prevedono un range più ampio. Questo potrebbe disturbare il funzionamento di alcuni programmi rigidamente legati a particolari configurazioni hardware di sistema. Dovrebbe invece sopravvivere il software che usa vettori di sistema anziché comunicare direttamente con l'hardware. Questo significa che molti giochi non funzioneranno correttamente col nuovo chip set.

Altri cambiamenti riguardano il controllo del pennello elettronico, tre bit aggiuntivi di risoluzione verticale, un nuovo registro di controllo del blitter che scrive solo sugli otto byte più bassi (per rendere più veloci le applicazioni) e la possibilità di un coprocessore matematico per il blitter.

Il nuovo chip Denise ha un bit extra di risoluzione orizzontale, il controllo sul blanking composito esterno, e rende disponibile il nuovo modo superhires. Si tratta di un display a due bit plane 640 x 400 (640 x 512) privo d'interlace. Richiede un monitor in grado di offrire una larghezza di banda di 31,5 KHz. Un monitor multisync dovrebbe bastare, ma la Commodore ha intenzione di mettere a disposizione un monitor bi-sync che potrà dare supporto al display standard (15,75 KHz di larghezza di banda) e contemporaneamente al modo superhires. Si configurerà automaticamente per dare il corretto supporto a ogni modo display, e dovrebbe costare meno di un monitor multi-sync.

Il nuovo monitor A2024 permette una risoluzione senza flickering di 1008 x 1024

Il nuovo chip Denise mette ancora a disposizione il modo half-brite. Le rilevazioni di collisioni non funzionano nel modo superhires e il numero degli sprite è limitato a quattro. Sono in grado di funzionare anche sprite di tipo no-attached.

Software sistema e linguaggi

Ci saranno alcuni cambiamenti nel menu Preferences e nel *Workbench*. In Preferences non è stato lasciato abbastanza spazio per le nuove caratteristiche che la Commodore vuole introdurre nel sistema operativo. Il *Workbench* sarà modificato in modo da poter utilizzare direttamente tutti i comandi CLI. Quindi occorre fare spazio alle nuove voci che andranno a riempire la struttura dei menu. Inoltre il *Workbench* supporterà un numero variabile di bit plane.

Il serial.device e il parallel.device saranno riscritti. Attualmente non permettono di aggiungere al sistema altre porte seriali o parallele. L'attuale dispositivo seriale inoltre non è adatto a tutte le possibilità di funzionamento in rete.

Verrà aggiunto anche l'overscan di sistema. Ora come ora, i programmatori sono costretti a creare un supporto per l'overscan nel loro spazio privato di Intuition.

La Commodore riconosce la necessità dell'overscan per le applicazioni video, ma vuole evitare che ciascun programmatore segua una propria tecnica personale per ottenerlo. Le attuali tecniche non saranno impedite, ma sarà comunque installato un metodo di overscan "ufficiale".

L'AmigaDOS verrà potenziato rendendo standard il fast file system, verrà dato supporto al fast file system anche sui floppy disk, e finalmente ci sarà un requester standard per il file system. Ci sarà inoltre un più ampio set di chiamate DOS, chiusure di record per le appl

in rete, miglioramenti all'AmigaShell, e al linguaggio originale (incluso un comando DO... WHILE) e comandi DOS aggiuntivi.

Un gran numero di programmatori si è detto favorevole all'aggiunta di AREXX al sistema. AREXX è la versione per l'Amiga di REXX, un efficace macro-linguaggio usato per le comunicazioni tra processi. Consiste in un linguaggio di alto livello che rende più facile lo scambio d'informazioni tra programmi diversi, e mette a disposizione dell'utente un accesso per creare le proprie macro di comunicazione tra le applicazioni che supportano AREXX.

Un'affollata tavola rotonda sui problemi delle porte multiple ha trattato il tema dell'eventuale installazione di schede modem, schede per la connessione in rete, schede per l'acquisizione di dati, e ha fatto il punto sui problemi che procurano gli attuali driver per i dispositivi seriale e parallelo. Sull'Amiga, i dispositivi che si autoconfigurano hanno i driver situati nel cassetto Expansion e vengono inizializzati durante la chiamata a Binddrivers. In questo modo anche questi dispositivi si aggiungono alla DeviceList di sistema. Se due dispositivi hanno lo stesso nome (per esempio serial.device) il Port-Handler di sistema individuerà solo il primo nodo con quel nome.

Cambiare il nome del driver (chiamandolo modem.device, per esempio) è solo una soluzione provvisoria perché per l'applicazione che apre il dispositivo seriale non c'è modo di comunicare che il modem.device è solo un altro tipo di porta seriale disponibile per la ridirezione dell'output.

L'hard disk A590

La Commodore ha discusso i suoi progetti sul sistema operativo UNIX. Attualmente è in programma una nuova release dell'A2500UX per l'inizio dell'89. Per lo UNIX si tratterà del Sistema V, terza release, prima versione. Darà supporto alla gestione di memoria a pagine virtuali, con un Gigabyte di memoria. Saranno inclusi gli stream e l'RFS (remote network filesystem). I programmatori avranno accesso a un compilatore C ottimizzato, perfettamente compatibile con il 68020 e con i set d'istruzioni in virgola mobile del 6888X. Sarà messa a disposizione anche una finestra shell. È inoltre probabile che la Commodore introduca una versione BSD dello UNIX per il mercato universitario.

Si è parlato anche di un nuovo prodotto hardware, l'A590. Si tratta di un hard disk da 20 MB per l'A500. È un dispositivo SCSI DMA, dotato di un'interfaccia SCSI. Si connette direttamente al bus di espansione dell'A500 e non richiede altre interfacce. Benché sia difficile aver notizie dettagliate, sembra che la Commodore abbia deciso di produrre l'A590 con un drive da 3,5" da soli 20 MB. Tale quantità di memoria potrebbe essere aumentata, ma l'uso di chip 256 x 4 rende difficile ogni variazione, e sembra che l'intero set di chip verrà fornito direttamente dalla Commodore.

L'A590 fa uso di un Chip Custom DMA SCSI. Ci sono voluti due anni per svilupparlo a West Chester. La potenza aggiuntiva dell'A590 è fornita da un'unità separata per mantenere al minimo lo sviluppo di calore; l'unità principale contiene anche una piccola ventola assiale. Speciali circuiti sono destinati a riconoscere se l'A500 è acceso o spento, e a mettere in funzione conseguentemente il trasformatore dell'A590. Non è stato annunciato né il prezzo né quando sarà disponibile al pubblico.

Una finestra della più recente versione del Preferences presente nel Workbench

Nasce il nuovo Workbench 1.4

C'è stato un interessante scambio di opinioni sulla direzione che dovrebbe prendere la prossima versione (V1.4) del sistema operativo. È stato assicurato il supporto per l'ECS e per il monitor A2024. Si è parlato di progetti per un console device "character mapped" che permetterebbe di non perdere informazioni presenti nelle finestre CLI e Shell, e garantirebbe un buffer "storico" p...

non perdere caratteri usciti dalla finestra con lo scroll. Questo consentirebbe all'utente di rimpicciolire e ingrandire a piacere una finestra CLI con il conseguente riposizionamento del testo al suo interno. Permetterebbe inoltre uno scroll all'indietro per richiamare qualunque cosa sia stata presente nella finestra.

Verrà aggiunto il supporto per effettuare, tramite una chiamata di sistema, lo scaling dei disegni grafici (lo scaling è la correzione del rapporto tra le dimensioni di un disegno, necessaria perché il disegno non risulti deformato quando la risoluzione orizzontale è diversa da quella verticale. La correzione si ottiene moltiplicando una delle coordinate per un opportuno "fattore di scaling", n.d.r.). Verrà inoltre aggiunto un supporto per le fonti a colori e per un più rapido scroll del testo. Si stanno progettando anche directory di fonti multiple. Tutte queste caratteristiche saranno presenti in ROM.

I miglioramenti a Intuition permetteranno ai programmatori di creare gadget supplementari. Daranno la possibilità di avere gadget rotanti e dial. Verrà dato supporto a un Commodities Exchange, che permetterà alle applicazioni di comunicare con il sistema direttamente, tramite il Commodities Exchange, senza quindi dover installare un nuovo handler di input. Questo permette di evitare la creazione di un processo che si appropri dell'input proveniente dall'input handler di sistema per passarlo a un altro processo. In altre parole il sistema dà supporto ad applicazioni come *PopCLI, Mach* e *SunMouse*.

La soluzione che è stata proposta cambierebbe il modo con cui viene definito lo spazio del nome dell'Exec per permettere l'aggiunta di un ID al tipo della porta cosicché le applicazioni potranno ora chiamare SER:modem1 per usare una scheda modem. Questo richiederebbe un nuovo Port Handler per analizzare correttamente il nome e passarlo all'opportuno dispositivo Exec. L'aggiunta del numero dell'unità assicurerà che durante il processo di autoconfigurazione non siano presenti due o più porte con lo stesso ID. Usando un nome per le porte si evita di dover ricorrere a una codificazione hardware per ogni slot.

Un'altra area di discussione ha esaminato la possibilità che una specifica applicazione prenda il controllo dell'intero sistema, nel tentativo di ottenere i massimi risultati. È quello che fa di solito chi progetta i videogiochi. Ma ci si è trovati di fronte a un problema di non facile soluzione. Benché per alcuni utenti possa rivelarsi utile, per coloro che dispongono di un hard disk, che caricano un gran numero di utility all'atto dello startup o che disporranno eventualmente di un'unità connessa in rete, sarebbe inaccettabile la perdita dell'ambiente corrente per il caricamento del gioco e la conseguente necessità di dover eseguire il reboot. Dovrebbe quindi essere fornita la possibilità di prendere il controllo del sistema e ritornare al sistema senza cancellare le applicazioni esistenti, o almeno si dovrebbe informare l'utente che non è possibile iniziare il gioco senza perdere l'ambiente corrente.

Librerie personalizzate

Un seminario d'importanza "strategica" per i programmatori è stato dedicato alle librerie. Le librerie sono set di funzioni che possono essere utilizzate per un'ampia varietà di applicazioni. L'Amiga fa uso di diverse librerie che si trovano nella directory LIBS del *Workbench*. I programmatori che si servono dello stesso set di routine in molte situazioni diverse possono risparmiare una grande quantità di memoria e di tempo raccogliendo le loro routine in un'unica libreria. Una parte della proposta AWG comprende la creazione di nuove librerie che possano essere utili a molte applicazioni.

Nel corso del seminario sui file di sistema è diventato chiaro che ora è possibile realizzare e montare un filesystem che permetta all'Amiga di leggere e scrivere dati su disco in altri formati, compreso l'MS-DOS. Non dovrebbe trattarsi di un'utility per convertire i file, ma dovrebbe essere parte integrante dell'Amiga. Si potrà per esempio configurare il drive df1 in ambiente MS-DOS, e automaticamente ogni dato scritto tramite quel drive risulterà in formato MS-DOS.

Un convegno "notturno"

Nel corso del convegno si sono tenute moltissime riunioni informali. Curiosando in giro si potevano scoprire discussioni che si protraevano fino a sera, sui relativi meriti dei compilatori C Lattice e Manx, sui migliori editori di testo, e sugli orari di lavoro dei programmatori.

La Commodore ha messo a disposizione una "sala hacker", cioè una stanza piena di Amiga che veniva lasciata ai convenuti verso sera, non appena si chiudeva la parte "ufficiale" del convegno, che restava aperta fino alle cinque di mattina. La stanza era sempre affollata, e lì i programmatori facevano bella mostra delle loro più recenti creazioni, lavoravano sulle idee nate durante le discussioni della giornata e si cimentavano nei giochi più famosi fino a notte fonda.

Ogni partecipante ha ricevuto in omaggio una maglietta ricordo, un fascicolo con tutti i dati e le informazioni sul convegno e un set di dischi contenenti codici sorgente per tutti gli esempi di programmazione discussi nel corso degli incontri. È stata inclusa anche una nuova release "gamma" del sistema operativo. Se tra i lettori c'è qualche programmatore che è interessato allo sviluppo di software per l'Amiga dovrebbe cercare di procurarsi una copia degli atti del convegno e dei dischi (Commodore Technical Support, 1200 Wilson Dr., West Chester, PA 19380, USA). Saranno un aiuto prezioso.

I PC COMMODORE IBM COMPATIBILI DELLA TERZA SERIE

La terza serie dei PC IBM compatibili della Commodore comprende molti dispositivi, come il video controller per la grafica e il controller per l'hard disk, che in genere sono disponibili solo come optional

di Luca Giachino

Aleggia nell'aria un inaspettato riflusso, che un po' è risveglio dall'assopito fervore, un po' è progresso tecnologico. I protagonisti di quest'ondata di ritorno, cioè il PC IBM compatibile e il sistema operativo MS-DOS, sono ormai delle pietre miliari nel mondo informatico, ruolo che giustamente detengono per la svolta che hanno imposto al mercato dei computer e per la loro incredibile longevità.

Queste considerazioni danno lo spunto per un'analisi storica del personal computer che ci riporta a quasi una decade fa. Era il momento in cui venivano presentati i nuovi PC IBM, e i computèr diventavano un fenomeno di massa molto più consistente che in passato. Era l'immediata affermazione di uno standard che aveva convinto subito tutti: il sistema operativo MS-DOS.

Ma c'è una seconda considerazione da fare: i PC IBM e l'MS-DOS sono ormai strumenti di lavoro rudimentali e obsoleti. Eppure oggi, dopo dieci anni – che per l'informatica valgono quanto mezzo secolo – assistiamo a un fenomeno di riscoperta dei PC IBM compatibili che sembrerebbe darci torto... e non si tratta di una corrente di origine orientale *made in Taiwan*, ma di una rinascita d'interesse sostenuta dalle grandi firme del mercato informatico occidentale.

Osservando il mercato, si possono individuare senz'altro almeno due fattori che stanno alle spalle di questo revival. Il primo riguarda la trasformazione d'immagine subita negli ultimi anni dal PC: da strumento altamente professionale e costoso, il personal è diventato agli occhi del pubblico un computer economico e adatto anche a coprire ruoli più ludici. Effettivamente, agli albori della loro storia i PC rappresentavano una spesa non indifferente, e alcune applicazioni raggiungevano costi che potevano essere ammortizzati solo in ambiti professionali; inoltre i programmi erano caratterizzati da una tale serietà che difficilmente sarebbe stato possibile inquadrare il PC in aree applicative diverse da quelle professionali.

Oggi, invece, moltissime persone cominciano a vedere i PC sotto una nuova luce. Sicuramente determinante è stato l'abbassamento vertiginoso dei prezzi, che ha permesso la diffusione a macchia d'olio dei PC sulle scrivanie di casa e non più solo negli uffici. La diretta conseguenza di questo fenomeno è stata la trasformazione stessa del software, passato da una produzione estremamente seria e professionale a una presenza sempre più diffusa di programmi d'intrattenimento e di videogame, che ha dato origine a un'ulteriore spinta verso la metamorfosi del ruolo dei PC.

Il secondo aspetto di questa situazione affonda le sue radici anche nell'innovazione tecnologica, che ha permesso ai costruttori di ridurre le dimensioni dei PC e aumentarne le qualità. Oggi un PC IBM compatibile, oltre all'inalterata ossatura che reggeva anche i primi modelli, viene normalmente dotato di molte caratteristiche che nel passato avrebbero richiesto l'acquisto extra di schede ben poco economiche, pur essendo attestato su prezzi base molto più bassi. I due prodotti che illustriamo in questo articolo, il PC 10-III e il PC 20-III della Commodore, sono ottimi rappresentanti di questa situazione "di riflusso" che ancor oggi, nel momento in cui si affermano nuovi standard operativi e nuovi processori, spinge un colosso come la Commodore a migliorare la sua linea di PC MS-DOS per offrire prestazioni sempre più elevate a costi sempre più concorren-

I PC 10/20-III

Il PC 10-III e il PC 20-III sono due computer la cui unica differenza riguarda l'organizzazione della memoria di massa. Il PC 10-III è dotato di due unità disk drive da 360K per dischi da 5,25", mentre il PC 20-III anziché il secondo disk drive contiene un hard disk da 3,5", da 20 MB. Dal momento che questa è l'unica differenza fra le due macchine, abbiamo eseguito la prova unicamente sul PC 10-III. Le considerazioni che faremo valgono quindi anche per il PC 20-III.

Il primo particolare che salta all'occhio guardando il PC 10-III riguarda le dimensioni del cabinet, molto più stretto di quello presente nei computer della seconda serie. La profondità è rimasta invariata (39 cm), l'altezza è maggiore di mezzo centimetro (15 cm), ma la larghezza è minore di ben 14 centimetri (35,5 cm). È senz'altro un particolare degno di nota se teniamo presente quanto erano inutilmente ingombranti i vecchi cabinet.

Il nuovo PC è di aspetto gradevole, il colore dominante è il classico beige e il design richiama quello della seconda serie. Sul lato frontale risiedono i due disk drive da 5,25", dotati della comoda levetta per l'abbassamento delle testine, e due led per indicare l'accensione della macchina e l'attività dell'hard disk. Sotto il pannello frontale si trovano alcune feritoie d'aerazione che contribuiscono all'aspetto esteticamente gradevole della macchina.

Sopra: il PC 10-III funziona in modo grafico con l'aggiunta di una scheda grafica. Sotto: le dimensioni dei PC 10 della seconda e della terza serie a confronto

Sul lato destro dell'unità centrale risiedono un comodissimo tastino di reset, grazie al quale è possibile evitare di allungare il braccio fino all'interruttore centrale disposto sul lato posteriore della macchina, e la presa DIN pentapolare per il collegamento della tastiera.

Sul lato posteriore del cabinet si trovano l'interruttore centrale, la presa d'aerazione per la ventola di raffreddamento (poco rumorosa, come quelle della seconda serie), la presa d'alimentazione, quattro feritoie in corrispondenza degli slot interni, una porta seriale RS-232 e una parallela Centronics, una presa maschio DB-9 per il collegamento del mouse Microsoft e compatibili, una presa d'uscita per il segnale video composito e una per il segnale RGBI. Accanto alle prese video si trova una finestrella che consente l'accesso a un gruppo di switch, destinati a selezionare il tipo di output video desiderato (a seconda del monitor di cui si dispone). È possibile collegare un monitor con ingresso composito,

RGBI o TTL.

Ogni presa è facilmente riconoscibile grazie all'etichetta che la contraddistingue e la descrive. Troviamo inoltre una comodissima descrizione delle combinazioni di switch più comuni, grazie alla quale non siamo costretti a consultare ogni volta il manuale operativo.

Il PC 10-III è dotato di una tastiera italiana avanzata da 101 tasti, pienamente compatibile con quella standard IBM installata nei PC AT. Ergonomicamente è comoda, i tasti sono ben distribuiti e facilmente accessibili. Sul lato superiore destro sono disposti tre led che documentano lo stato dei tasti toggle Num Lock, Caps Lock e Scroll Lock. Rispetto alla tastiera non avanzata, la posizione di questi tre led è ora molto più comoda, e consente di rilevare lo stato dei tasti con un semplice colpo d'occhio.

Sotto la tastiera sono presenti due alette che permettono di inclinarla verso l'utente di qualche grado. Consentono però una sola alternativa alla posizione normale, mentre le alette in dotazione alla seconda serie ne consentivano due.

Il cavo di collegamento con l'unità centrale è a spirale e può essere disposto lungo la tastiera inserendo la parte non spiralizzata all'interno di una feritoia che scorre lungo tutto il lato posteriore della macchina.

Nell'insieme la tastiera appare robusta e non smentisce la fama delle tastiere Commodore, caratterizzate in genere da un'ottima risposta allo sfioramento dei tasti.

Sul lato superiore del cabinet può risiedere il monitor. La Commodore prevede per la vendita due diverse configurazioni: quella di base include il monitor Commodore 1402 monocromatico a fosfori verdi da 12", mentre la configurazione PC 10/20-III C include il monitor Commodore 1084 a colori da 14". Entrambi i monitor sono di ottima qualità, e scegliere l'uno o l'altro dipende unicamente dall'interesse dell'acquirente per le potenzialità grafiche dei PC terza serie, su cui ci soffermeremo più avanti.

Il PC 10-III è dotato di tre manuali. Uno, in lingua inglese, è il manuale operativo, nel quale sono raccolti in modo chiaro i dati tecnici della macchina, la procedura di avviamento in ambiente MS-DOS, le caratteristiche grafiche, l'installazione dell'hard disk... Gli altri due, in lingua italiana, sono il manuale dell'MS-DOS in dotazione alla macchina (versione 3.21) e il manuale del GW BASIC (versione 3.21). La traduzione in italiano dei manuali dovrebbe rendere felici tutti gli utenti che pretendono (e non hanno torto) documentazioni facilmente comprensibili quando fanno un investimento in un nuovo sistema.

Sopra: il PC 10-III. Sotto: la parte posteriore del PC 10-III; si notino le porte per mouse, segnale videocomposito e segnale RGBI, gli switch, le porte RS-232 e Centronics

Le caratteristiche tecniche

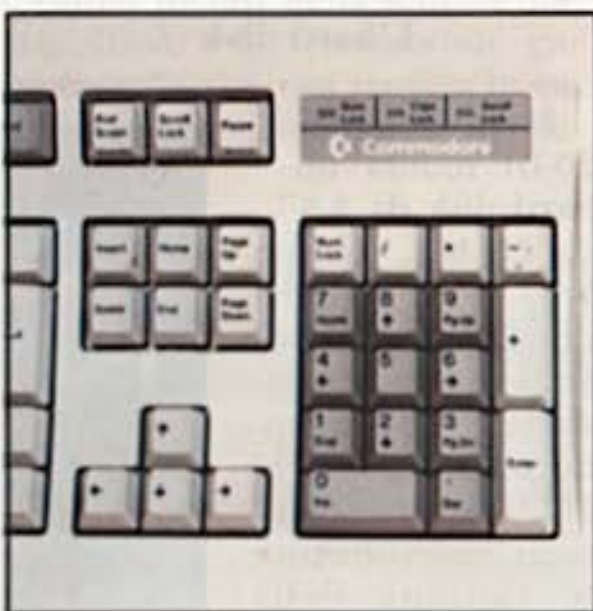
Il microprocessore, è l'Intel 8088, 16 bit, che il PC 10-III consente di far funzionare a tre diverse frequenze di clock: 4.77 MHz, 7.16 MHz e 9.54 MHz. La frequenza di clock è selezionabile in tre modi distinti. Tramite tre combinazioni di tasti CTRL, tramite l'apposito comando SPEED del DOS, e da programma. In questo modo l'utente ha la possibilità di avviare la macchina alla frequenza di clock che ritiene opportuna inserendo il comando SPEED nel file AUTOEXEC.BAT che realizza la startup sequence del sistema. Se l'applicazione che si manda in esecuzione non è compatibile con la frequenza di clock impostata, è sufficiente intervenire con l'opportuna combinazione di tasti CTRL. Il PC 10-III è in grado di accogliere anche il coprocessore matematico 8087,

particolarmente utile per tutte quelle applicazioni che sfruttano a fondo la grafica.

La memoria RAM di serie in dotazione ai PC terza serie è da 640K, la massima indirizzabile dal microprocessore 8088. I nuovi utenti non dovranno più chiedersi se è il caso di espandere la memoria o no, quanto costa e se possono sorgere dei problemi: il PC 10-III offre già la massima memoria installabile negli XT compatibili.

Il BIOS di cui è dotata la macchina è il classico Phoenix, quindi è pienamente compatibile con il BIOS XT. In particolare, questa versione, la 4.35, è in grado di autoconfigurarsi, cioè riconosce le espansioni collegate durante la fase di startup e configura automaticamente i relativi indirizzi di I/O. Grazie a questa qualità non è necessario imporre alcun comando per rendere attive le espansioni.

Nel PC 10-III risiedono diversi dispositivi che su altri PC sono disponibili solo come schede d'espansione, e quindi comportano costi aggiuntivi. Nel PC 10-III sono montati di serie sulla scheda madre, lasciando così liberi gli slot d'espansione per ricevere altre schede, ed evitando all'utente di dover aggiungere espansioni per ogni prestazione in più che desidera.

Questi dispositivi di serie sono un chip video controller di design avanzato in grado di gestire la grafica MDA, CGA, Hercules e Plantronics; un'interfaccia SCSI per controllare un hard disk compatibile XT (questo controller è presente in entrambi i modelli, ma solo il PC 20-III è dotato anche dell'hard disk di serie), un'interfaccia per il collegamento del mouse Microsoft (o il compatibile Commodore 1352), una porta parallela Centronics e una seriale RS-232, un orologio/calendario con batteria tampone. Praticamente c'è tutto, o quasi, per soddisfare le normali esigenze di una larghissima fascia di utenti. Non dimentichiamo comunque che per i più esigenti sono disponibili tre slot di tipo XT per accogliere schede d'espansione opzionali.

Sopra: un particolare della tastiera. Sotto: FS in funzione con un monitor a colori

La grafica

Come abbiamo già sottolineato, i PC terza serie montano sulla scheda madre un video controller in grado di emulare la grafica MDA, CGA, Hercules e Plantronics. Sebbene con questo controller sia possibile eseguire programmi grafici come *Windows* e *Flight Simulator* senza dover annotare nel bilancio la spesa di una scheda grafica aggiuntiva, per poter effettivamente usufruire della grafica è necessario disporre del monitor Commodore 1084 a colori da 80 colonne, dotato di ingresso RGBI, incluso nelle configurazioni PC 10/20-III C. Infatti il video controller montato sulla scheda madre purtroppo non consente di visualizzare la grafica sul monitor Commodore 1402 monocromatico.

Nei PC seconda serie la grafica si ottiene tramite una scheda di serie installata in uno degli slot. Grazie a questa scheda, con un'opportuna configurazione degli switch, si possono avere effetti grafici anche con il monitor Commodore 1402 monocromatico. In questo modo l'utente può disporre della grafica monocromatica senza dover espandere il sistema, e quindi senza effettuare spese addizionali. Purtroppo, e questa ci sembra una scelta poco felice, con la terza serie questa possibilità non è più disponibile e l'utente è costretto a procurarsi un monitor a colori.

Nelle nostre prove siamo riusciti a visualizzare la grafica di *Flight Simulator* impiegando il monitor a colori 1084 e configurando opportunamente gli switch per la selezione (situati sul retro della macchina e accessibili dall'ester-

no). L'operazione si è rivelata molto semplice, grazie anche alla chiarezza del manuale.

Ma eseguendo questa prova eravamo molto delusi dall'impossibilità di accedere alla grafica con il normale monitor monocromatico Commodore 1402. Per curiosità, abbiamo provato a escludere il video controller di serie, e abbiamo prelevato da un PC della seconda serie la scheda grafica, per inserirla in uno degli slot all'interno del PC 10-III (avendo cura di collegarlo al monitor monocromatico). Leggendo il manuale ed eseguendo qualche tentativo, abbiamo individuato la configurazione per gli switch del PC 10-III che permette di escludere il video controller di serie e attivare la scheda grafica inserita in uno slot. Tutto ha funzionato perfettamente, consentendoci di accedere alla grafica anche con il monitor monocromatico 1402. Inoltre, la possibilità di escludere il video controller di serie denota la buona flessibilità di questa macchina.

Riportiamo qui di seguito la configurazione degli switch della scheda grafica e del PC 10-III.

Scheda grafica:
gli switch 1,2,4,5,6 devono essere in posizione OPEN.

PC 10-III:
gli switch sul retro del computer devono essere tutti a OFF tranne lo switch 4.

Quindi, chi non desiderasse acquistare il monitor a colori 1084, può procurarsi la scheda grafica montata sui PC seconda serie e risolvere il problema, anche se in modo forse un po' macchinoso.

Nella prova abbiamo sottoposto il PC 10-III a molte ore di continuo funzionamento, sostituendolo a uno di quelli che impieghiamo normalmente in redazione. Abbiamo avuto modo di provarlo con diversi pacchetti applicativi, focalizzando l'interesse su quelli che impiegano la grafica, come *Flight Simulator* e *Lotus 1-2-3*. Non abbiamo però avuto modo di provare il funzionamento dell'interfaccia per il mouse Microsoft.

La macchina si è dimostrata affidabile, e soprattutto conforme alle caratteristiche dichiarate dal costruttore.

L'hard disk

Mentre il PC 20-III monta un hard disk da 3,5", da 20 MB, di tipo XT, il PC 10-III è dotato solo del controller per comandarlo. Aggiungendo un hard disk al PC 10-III si aumentano notevolmente le capacità della macchina, ottenendo praticamente un PC 20-III.

Sopra e sotto: l'interno del PC 10-III con il telaio metallico montato e smontato

Discutere quindi l'aggiunta al PC 10-III di un hard disk può sembrare inutile, ma se teniamo presente che spesso certi limiti del proprio computer si evidenziano solo con il tempo, per le accresciute esigenze dell'utente, può facilmente nascere il desiderio di trasformare il proprio PC 10-III in un PC 20-III.

Vediamo quindi come gli utenti del PC 10-III possono orientarsi per dotare la macchina di un hard disk.

Se la scelta cade su un qualunque hard disk da 5,25", si rende necessario occupare uno slot con il relativo controller ed eliminare un disk drive per sostituirlo con l'hard disk: non è certo la soluzione migliore, ma almeno è una soluzione possibile.

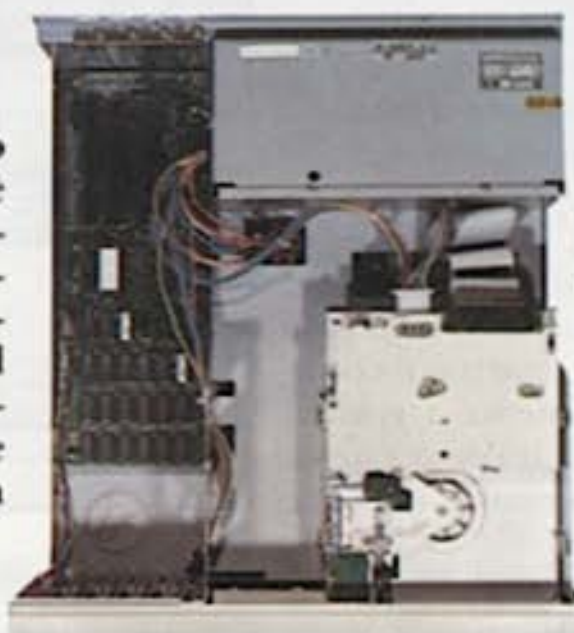
Invece con un hard disk da 3,5", di dimensioni più piccole del precedente, si occupa uno slot per il controller e si colloca l'hard disk a fianco dei due disk drive serrandolo all'apposito sostegno metallico. In questo modo si può disporre dell'hard disk senza dover eliminare uno dei due disk drive. Per esempio, il PC 20-III viene distribuito con un disk drive e l'hard disk da 3,5", ma all'occorrenza è possibile espanderlo con un secondo disk drive senza rimuovere l'hard disk.

Ma la scelta migliore consiste nell'impiegare un hard disk da 3,5", da 20 MB, XT compatibile, e collegarlo al controller montato di serie nei PC terza serie. In questo modo non viene comunque occupato nessuno slot e si risparmia spazio per altre espansioni. Inoltre, dal momento che la macchina dispone di un controller di serie, doverne acquistare uno nuovo sarebbe assurdo.

Nel manuale in dotazione alla macchina viene comunque indicato quali jumper interni devono essere modificati per disattivare il controller fornito con il computer.

L'interno

Accedere all'interno del cabinet è relativamente semplice. È sufficiente svitare sei viti ed estrarre il coperchio. Nello svol-

gimento di questa operazione bisogna aver cura di tenere premuto il tastino di reset laterale, che altrimenti ostacolerebbe lo scorrimento del cabinet.

All'interno, il PC terza serie si presenta pulito e ordinato. Spicca subito per le dimensioni non trascurabili l'alimentatore della macchina. Il costruttore dichiara che questo alimentatore è in grado di fornire una potenza sufficiente all'intera macchina, compresi due disk drive, un hard disk e tre schede inserite negli slot. Nella nostra prova non l'abbiamo dotato di tutti questi dispositivi, ma l'abbiamo tenuto sotto carico per molte ore senza notare surriscaldamenti anomali.

La scheda madre è parzialmente nascosta dal telaio metallico che regge i due disk drive. Sopra il telaio c'è abbastanza spazio per collocare un hard disk da 3,5", e sono stati predisposti gli appositi fori.

Estraendo il telaio, operazione possibile solo se è stato rimosso il pannello frontale della macchina, si scopre l'intera scheda madre, che appare di buona qualità. Le eprom sono montate su opportuni zoccoli che ne consentono la facile sostituzione nel caso vengano distribuite nuove release del BIOS. L'assemblaggio dei componenti è ben curato, ma non mancano alcuni collegamenti posticci "dell'ultimo momento".

Sulla scheda madre trovano posto anche i tre slot XT compatibili per l'aggiunta di eventuali schede d'espansione.

Conclusioni

La nuova serie dei PC Commodore ci ha favorevolmente impressionato. La politica adottata per queste macchine può essere descritta in poche parole: affidabilità, dimensioni ridotte, capacità grafiche, prezzo concorrenziale. Difficile chiedere di meglio.

Le molte capacità che offrono già nella versione standard dovrebbero essere sufficienti alla maggior parte degli utenti, e solo in casi particolari dovrebbe nascere l'esigenza di un'espansione.

Ci è piaciuta la documentazione in italiano sul DOS e il GW-BASIC, ma troviamo spiacevole l'obbligo di ricorrere a un monitor a colori per servirsi della grafica.

I prezzi di queste macchine sono decisamente contenuti, soprattutto tenendo conto dei dispositivi inclusi nella scheda madre. Inoltre la Commodore garantisce l'assistenza tecnica in caso di malfunzionamenti o problemi di ogni tipo.

Ci sembra quindi che la terza serie possa figurare più che degnamente in questa frequentatissima arena rappresentata dal mercato dei PC IBM compatibili. Non sono cosa da poco alcuni piccoli miglioramenti di una base già solida come l'MS-DOS (tanto solida da avere alle spalle dieci anni di successi) a prezzi quattro o cinque volte inferiori a quelli del passato. ■

DATI TECNICI

Microprocessore:	Intel 8088, frequenza di clock selezionabile fra 4.77 MHz, 7.16 MHz e 9,54 MHz coprocessore matematico 8087 opzionale
Memoria RAM:	640K di serie
Monitor:	Commodore 1402 monocromatico per i PC 10/20-III Commodore 1084 a colori, 80 colonne, alta risoluzione per i PC 10/20-III C
BIOS:	pienamente XT compatibile, capacità di autoconfigurarsi secondo le espansioni inserite
Grafica:	chip video controller avanzato, compatibile con la grafica MDA, CGA, Hercules e Plantronics
Memoria di massa:	PC 10-III: due disk drive da 360K per dischi da 5,25" PC 20-III: un disk drive da 360K per dischi da 5,25" e un hard disk da 3,5", da 20 MB
Controller:	controller per hard disk XT compatibili incluso sulla scheda madre
Espandibilità:	tre slot XT compatibili per le schede d'espansione
Dispositivi:	controller per hard disk XT orologio/calendario con batteria tampone interfaccia per mouse Microsoft e compatibili porta seriale RS-232 porta parallela Centronics
Tastiera:	italiana, avanzata AT compatibile, 101 tasti
Dimensioni:	15 x 39 x 35.5 cm
Prezzi al pubblico IVA esclusa:	PC 10-III L. 1.490.000 PC 10-III C L. 1.890.000 PC 20-III L. 2.240.000 PC 20-III C L. 2.640.000

Il PC 60-40/80

Nell'arena dei PC altamente professionali la Commodore non poteva mancare troppo a lungo. Sulla scia dei PC 40 AT compatibili, appartenenti alla fascia 80286, ha prodotto una nuova linea di computer AT compatibili dotati di microprocessore 80386, 32 bit, l'ultimo nato e il più "dotato" della famiglia Intel.

La nuova linea comprende due modelli, il PC 60-40 e il PC 60-80. Il primo è dotato di un hard disk da 40 MB, mentre il secondo di un hard disk da 80 MB e l'interfaccia per il mouse Microsoft.

Esternamente il PC 60-40/80 mantiene la stessa linea del PC 40. Sul pannello frontale si trovano la serratura, con la quale è possibile disattivare la tastiera, il disk drive da 1,2 MB per dischi da 5,25", e due led che segnalano l'accensione della macchina e l'attività dell'hard disk.

Il cabinet misura 53 cm di larghezza, 42 cm di profondità e 15,5 cm d'altezza. Sul retro risiedono la presa di alimentazione principale e quella per alimentare il monitor, l'interruttore d'accensione centrale, due porte parallele Centronics e due seriali RS-232C (DB-9), il connettore per la tastiera e la porta per la connessione al monitor della scheda grafica EGA Wonder di serie.

La macchina è dotata di un monitor ad alta risoluzione (1000 x 800) da 14" e a fosfori verdi. È gradevole esteticamente, e dispone di un comodo sostegno mobile che consente di orientarlo secondo le proprie esigenze. Viene distribuita con tre manuali, di cui due, in italiano, documentano il sistema operativo MS-DOS e il GW-BASIC, mentre il terzo, in inglese, illustra i dati tecnici e le procedure della macchina, come ad esempio l'installazione. Oltre a queste pubblicazioni è allegato un manuale in inglese sulla scheda grafica EGA Wonder.

Il PC 60-40/80 funziona a due frequenze di clock, 8 e 16 MHz, selezionabili tramite il comando SPEED386, e consente d'installare opzionalmente il coprocessore matematico 80387, molto utile nel caso si impieghi la macchina in ambito grafico-scientifico, dove la quantità dei calcoli da eseguire è molto elevata.

La memoria RAM è da 2,5 MB, dei quali 512K sono forniti dalla scheda madre, mentre i restanti 2 MB derivano dalla scheda d'espansione installata su tutti i modelli.

La memoria di massa è ripartita fra il disk drive da 1,2 MB per dischi da 5,25", e l'hard disk interno da 40 MB. È possibile aggiungere un secondo disk drive e un altro hard disk, dal momento che il controller Western Digital della macchina, montato su un apposito slot d'espansione, è in grado di pilotare due disk drive da 1,2 MB e due hard disk.

Il sistema operativo in dotazione alla macchina è l'MS-DOS V3.21, che viene fornito già installato sull'hard disk, oltre che su disco, in modo che si possa eseguire il boot senza dover ogni volta inserire il disco del DOS. Per il PC 60-40 è possibile richiedere opzionalmente il sistema operativo MS-DOS Windows 386, mentre con il PC 60-80 viene fornito di serie insieme al mouse. Il BIOS su ROM è il Phoenix 80386 versione 1.00 C.

Il PC 60-40/80 dispone di due porte parallele Centronics e di due porte seriali RS-232C. La seconda porta parallela e la seconda porta seriale sono montate su una scheda inserita in uno slot. Oltre a questa espansione, il PC 60-40/80 dispone della scheda grafica EGA Wonder, che abbiamo già citato, e di un orologio/calendario dotato naturalmente di batteria tampone.

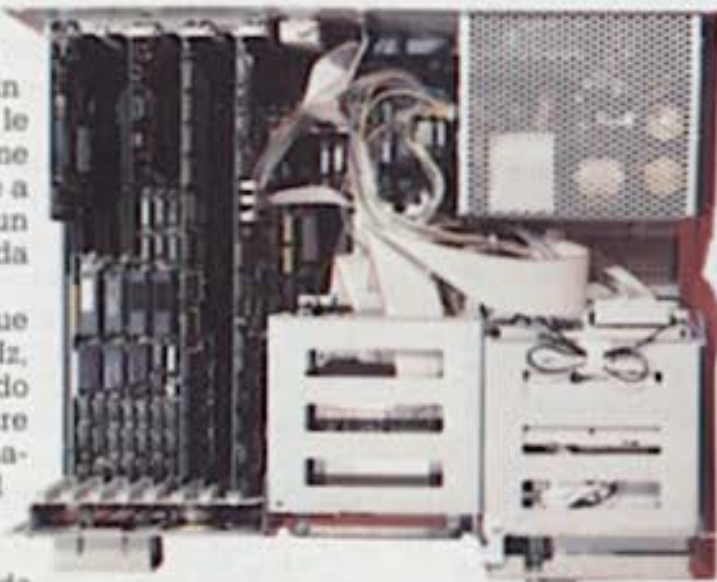
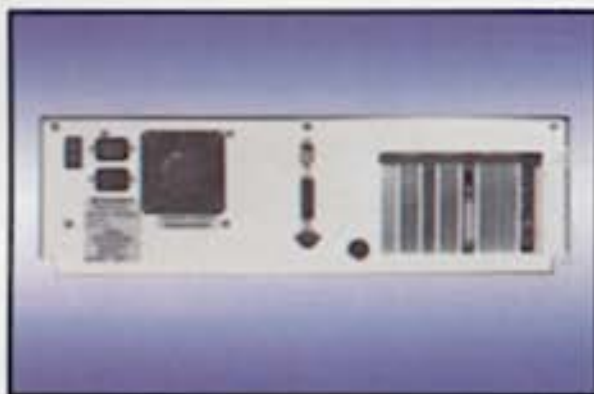
Sulla scheda madre si trovano tre slot AT (uno impiegato dalla scheda EGA), due slot XT (uno impiegato dalla scheda che contiene le porte parallela e seriale), due slot per contenere schede d'espansione RAM (uno occupato dalla scheda da 2 MB di serie) e uno slot AT occupato dalla scheda controller.

La tastiera, di marca Cherry, è il modello esteso compatibile AT e viene distribuita nela versione italiana. Si presenta abbastanza robusta. Può essere inclinata tramite due appositi sostegni estraibili. I tasti sono piacevoli da usare e poco rumorosi.

All'interno, il PC 60-40/80 appare ordinato, evidenziando un'ottima organizzazione dello spazio disponibile. Il telaio metallico di sostegno delle unità a dischi può contenere fino a due disk drive e due hard disk. Sulla scheda madre sottostante le eprom più importanti sono montate su zoccoli che ne consentono la facile sostituzione. Spiccano la quantità di slot e le quattro schede montate di serie: la scheda EGA, l'interfaccia parallela/seriale, l'espansione RAM da 2 MB, e il controller per le unità a disco. Il cablaggio interno della scheda madre, del telaio di sostegno delle unità a disco, e dell'alimentatore è stato svolto con cura e denota la qualità hardware del prodotto.

Queste due macchine possono apparire molto costose agli utenti di personal computer XT compatibili, ma bisogna considerare che con questi prodotti la Commodore vuole rivolgersi all'utenza professionale più esigente (AT), che desidera elevate prestazioni e affidabilità.

Prezzo al pubblico IVA esclusa:
PC 60-40 L. 8.490.000
PC 60-80 L. 9.990.000

LE FRONTIERE DELLA TELEMATICA

UNA RETE TELEMATICA SUL MONDO

Le telecomunicazioni digitali hanno assunto un ruolo fondamentale nello scambio di dati fra terminali, banche dati e servizi telematici. Le reti a commutazione di pacchetto offrono agli utenti di tutto il mondo collegamenti di elevata qualità a costi ridotti

di Simone Majocchi

L'avvento di un nuovo mezzo di comunicazione ha sempre contribuito a trasformare i costumi, le abitudini e i ritmi lavorativi. Nel nostro secolo la comunicazione ha ricevuto una spinta poderosa rispetto al passato: l'incredibile sviluppo tecnologico che ha caratterizzato il nostro periodo storico ha trasformato radicalmente la società, abbattendo molte barriere naturali, come le lunghe distanze e gli oceani. Protagonista indiscussa di questo rapidissimo processo evolutivo è certamente la telecomunicazione, il "cavallo alato" che oggi ci consente di comunicare in ogni momento con qualunque località del globo. Lo scambio d'informazioni a distanza in tempi brevi è diventato una necessità alla quale sono stati rivolti sforzi tecnologici senza precedenti

Oltre allo sviluppo dei sistemi di telecomunicazione verbale (come il telefono e la radio), alla maggiore e più rapida diffusione delle comunicazioni cartacee (come i periodici e le lettere), e all'avvento della comunicazione visiva (come la televisione e la fotografia), è nata verso la metà del nostro secolo la crescente esigenza di scambiare grandi quantità di dati in tempi brevi. Come conseguenza diretta della diffusione a macchia d'olio subita dai computer negli ultimi quindici anni si è sviluppata una nuova tecnica di trasmissione dei dati: la comunicazione digitale. In seguito alla sua affermazione la diffusione dei dati e delle informazioni sull'intera superficie terrestre ha subito un'ulteriore crescita che solo mezzo secolo fa sarebbe stata difficilmente prevedibile.

Fino a non molte decine d'anni fa erano necessari giorni perché un'informazione giungesse da un punto all'altro del globo. In questi ultimi anni le trasmissioni via etere e le comunicazioni via cavo sono progredite notevolmente, e ora disponiamo di risorse estremamente efficienti e sofisticate.

Con lo sviluppo dell'informatica si è parallelamente trasformato il modo con cui le informazioni vengono trasferite e archiviate: oggi perfino la comunicazione verbale inizia a essere trasformata in comunicazione digitale. I grandi centri informativi stanno diventando il metodo più economico e rapido per la gestione e la diffusione delle informazioni. Le reti di comunicazione sono nel pieno di una trasformazione rivolta a migliorare il transito di dati codificati piuttosto che quello vocale.

Nel nostro Paese si è già sentito molto parlare dell'isola ottica che la SIP espone puntualmente da un paio d'anni alla Grande Fiera d'aprile a Milano, come esempio della struttura che nel futuro caratterizzerà le reti di comunicazione urbane ed extraurbane. Nei prossimi anni vedremo diffondersi apparecchi telefonici che non invieranno più la nostra voce con segnale analogico, ma la codificheranno in segnali digitali. Questi telefoni disporranno di un'interfaccia seriale che permetterà non solo la comunicazione vocale, ma anche il collegamento diretto di un computer.

In queste pagine viene illustrato come sia già possibile ricorrere a reti di comunicazione particolari per allargare gli orizzonti dei collegamenti ai calcolatori e alle banche dati, servendosi di servizi telematici ancora un po' misteriosi per la maggior parte degli utenti di home computer, ma davvero molto efficienti.

Le reti a pacchetto

Per la trasmissione dei dati, già da molti anni sono state definite le Raccomandazioni X.25 (termine ufficiale che indica un particolare standard) dal Comitato Consultivo Internazionale per la Telegrafia e la Telefonia (C.C.I.T.T.). Questo standard regolamenta la realizzazione e la gestione di un particolare tipo di reti: le reti a commutazione di pacchetto.

BIX, la banca dati statunitense di Byte, dà il benvenuto richiedendo l'identificazione

Grazie agli sforzi impiegati per giungere a questo standard, oggi è possibile l'interconnessione di tutte le varie reti nazionali, e si è giunti alla creazione di una vera rete telematica, equivalente alle normali reti telefoniche che coprono il globo.

La principale caratteristica della rete a commutazione di pacchetto è la capacità di gestire il flusso dei dati in forma digitale, e cioè in modo molto più efficace di quanto si possa fare con la trasmissione dei segnali analogici come la voce.

La tecnica utilizzata per la trasmissione è quella della segmentazione del flusso di dati in pacchetti di lunghezza prefissata; ciascun pacchetto viene intestato con un blocco che indica l'utente di partenza (mittente) e quello di destinazione (destinatario). Ogni pacchetto trasmesso in rete condivide le stesse linee dati con altri pacchetti. In pratica si può pensare a una rete autostradale in cui tutti gli autoveicoli viaggiano a velocità estremamente sostenuta: ai caselli di entrata il veicolo viene etichettato con la destinazione e la provenienza, dopodiché si immette nella rete autostradale, accelerando per raggiungere la velocità degli altri veicoli. Il segreto della rete sta infatti nella capacità di accelerare i singoli pacchetti sulle tratte comuni in modo che in istanti diversi gli stessi cavi di collegamento siano condivisi da pacchetti diversi. Per fare un esempio, il collegamento con il PAD può avvenire a 300 baud o a 1200 baud, ma quando le informazioni sono state convogliate in pacchetti all'interno della rete, la velocità alla quale vengono trasmesse diventa molto superiore, in modo da minimizzare i tempi di attesa dei numerosi pacchetti che sfruttano la stessa tratta.

Il flusso e l'indirizzamento dei pacchetti viene gestito dai nodi: elaboratori in grado di dirottare i pacchetti sulle giuste tratte di collegamento. La valutazione della tratta ottimale prende in considerazione il carico presente e le eventuali disfunzioni della rete.

Ogni pacchetto viene temporaneamente memorizzato nei nodi e inviato verso il destinatario quando si rende libera una possibilità di trasmissione; in questo modo aumenta la capacità di ricetrasmissione per singolo collegamento fra i nodi.

Il colloquio con la rete a commutazione di pacchetto può avvenire in due modi distinti: direttamente in X.25, o in X.28. Nel primo caso è il nostro elaboratore che tramite un'apposita interfaccia provvede a spezzettare il flusso di dati in pacchetti, mentre nel secondo caso dobbiamo collegarci, tramite un modem e la linea telefonica ordinaria, a un particolare elaboratore dedicato, detto Packet Assembler Disassembler (PAD), che provvede a spezzettare in pacchetti il flusso di dati da noi generato.

Quest'opera di segmentazione in pacchetti per ottimizzare il flusso dei dati all'interno della rete, fa sì che quando utilizziamo la rete a commutazione di pacchetto per collegarci a un altro elaboratore non instauriamo una connessione fisica e costante per tutto il tempo, ma creiamo un collegamento virtuale in cui solo ai due capi della linea si ha l'occupazione di due canali di comunicazione (impegno di una linea telefonica o di una porta), mentre all'interno della rete il flusso d'informazioni circola sulle tratte simultaneamente ad altri flussi.

Dal momento che il trasferimento dei vari pacchetti all'interno della rete avviene a velocità elevatissima (64 Kbit/s e più) mentre la velocità alla quale operiamo è modesta, tutte queste operazioni di memorizzazione, trasferimento e trasformazione in pacchetti, vengono svolte senza che sia possibile accorgersene tangibilmente.

Gli ingenti sforzi che sono stati impiegati per arrivare a uno standard non erano rivolti solo alla creazione di una rete dotata di meno linee di comunicazione; il vero scopo era la creazione

rete che potesse essere offerta agli utenti a costi estremamente contenuti.

In pratica – già adesso – l'onere economico riguarda solo il canone di abbonamento, il volume di dati ricetrasmessi e la durata del collegamento, con alcune variazioni a seconda della velocità della comunicazione.

Non entra più in gioco il fattore distanza: questa è la vera novità introdotta dalle reti a commutazione di pacchetto; non c'è differenza fra un collegamento da Milano a Roma e uno da Milano a Sidney (a parità di tempo di connessione e di dati trasmessi). È evidente che la possibilità di collegarsi con tutto il mondo a un costo molto inferiore a quello di una telefonata normale è un notevole punto di forza.

Anche se non si può ancora dire che le reti a pacchetto siano universalmente note, sono già molti gli utenti che si stanno avvicinando con interesse alla rete italiana a commutazione di pacchetto *ITAPAC*, per poter finalmente sfruttare le possibilità offerte dalla telematica in modo completo e soprattutto a costi ragionevoli.

La rete ITAPAC

In campo professionale la rete *ITAPAC* rappresenta uno strumento essenziale per lo scambio d'informazioni, ed è per questo motivo che attualmente la SIP preferisce assegnare le porte disponibili a utenti che ne facciano un uso professionale, piuttosto che ai privati. Va infatti notato che al momento la nostra rete nazionale dispone di un numero limitato di porte, con conseguente limitazione del tetto massimo di utenti. Non è quindi per malanimo verso il privato che la SIP, per la rete *ITAPAC*, privilegia attualmente l'utenza professionale piuttosto che quella privata.

Ma quanto costerebbe a un privato l'abilitazione al servizio *ITAPAC*? Per rispondere dobbiamo compiere una breve analisi delle tariffe che seguono.

Il canone di accesso mensile alla rete è:

L. 12.150	collegamento a 300 baud
L. 17.100	collegamento a 1200 baud

La tariffa al minuto per l'occupazione di una porta è:

L. 13,5	collegamento a 300 baud
L. 18	collegamento a 1200 baud

Oltre al canone d'abbonamento e al tempo di connessione, viene valutato anche il traffico di dati. Ogni pacchetto ha la lunghezza massima di 128 byte, e viene addebitato l'importo di L. 1,78 se il pacchetto contiene fino a 64 byte, o di L. 3,56 se supera i 64 byte. L'importo per il traffico dei dati è infatti basato su due segmenti da 64 byte per ogni pacchetto e anche se il pacchetto non viene riempito completamente, si paga per due segmenti.

Stabilito il collegamento con un altro utente della rete, scatta un altro addebito, quello per la connessione virtuale che ammonta a L. 6,8 al minuto più uno scatto di connessione di L. 30 (in queste tariffe sono esclusi gli addebiti degli scatti per l'impiego della linea telefonica). A questi costi si deve ancora aggiungere una quota una tantum per l'installazione e l'attivazione di L. 200.000, il canone annuale per sede di utente di L. 200.000 e L. 7.200 per ricevere il NUI. Operando un rapido calcolo, il costo annuale si può valutare in L. 412.400 per i collegamenti a 1200 baud e L. 353.000 per i collegamenti a 300 baud, con un costo aggiuntivo per il primo anno di L. 200.000. A questi costi fissi va poi ad aggiungersi il prezzo di ogni collegamento. Ripetiamo ancora, a questo proposito, che i costi di collegamento non hanno alcun legame con la distanza.

Nella confusione di tutte queste cifre può non essere del tutto chiaro qual è l'onere economico reale che dovrebbe sopportare un utente *ITAPAC*. In effetti le tariffe sono abbastanza complicate, ma nell'insieme sono più che ragionevoli se si considerano le distanze che possono essere coperte allo stesso prezzo.

Forse qualcuno avrà notato che è stata citata la sigla NUI... e avrà pensato che negli articoli sulla telematica c'è sempre qualche

Il collegamento a ITAPAC, la rete a pacchetto italiana, avviene tramite le porte

codice misterioso, che sembra pensato apposta per impedire una lettura chiara. Ma non è certo questo il nostro scopo: *ITAPAC* assegna a ogni utente un codice d'identificazione, il Network User Identifier (NUI), per poterlo riconoscere e provvedere quindi a un addebito dei vari canoni e tariffe. Data la sua importanza, è trattato come una password ed è solo tramite il suo corretto inserimento che l'utente può procedere al collegamento.

Se quanto è stato finora illustrato vi è chiaro, allora siete effettivamente pronti per collegare il vostro computer a una rete a pacchetto che vi metterà in contatto col mondo intero, consentendo l'accesso a servizi altrimenti impossibili.

Ogni utente collegato alla rete, in Italia o all'estero, è identificato da un numero detto Network User Address (NUA), ed è essenziale conoscere questo numero per instaurare il collegamento con il suo terminale. Nel caso di un utente estero è anche necessario conoscere il prefisso della nazione ed eventualmente il numero d'identificazione della sua rete nazionale. Esiste ovviamente un elenco dei "prefissi internazionali", ma è molto più difficile entrare in possesso dei NUA esteri.

Se ad esempio vogliamo collegarci con l'*Hostess*, il database inglese contenente l'elenco completo dei NUA della rete a pacchetto britannica, dobbiamo chiamare il numero telefonico corrispondente al PAD a noi assegnato, instaurare il collegamento premendo due volte RETURN e procedere con l'inserimento del NUI e del NUA come segue:

N<il vostro NUI>-023421920
100515<RETURN>

In questa frase
- N indica che state per digitare NUI
- – indica la fine del campo NUI
- 0 introduce il prefisso internazionale
- 234 è il prefisso per la rete a commutazione di pacchetto inglese
- 21920100515 è il NUA di *Hostess*
- RETURN pone termine all'inserimento

Se non ci sono problemi dovrebbe apparirvi entro qualche secondo la scritta "ACP:COM", e

```
N-023421920100515
ACP:COM
                ====== HOSTESS ======

Welcome to HOSTESS, the information service for British Telecom's
Public Data Network.

==================== NEWSFLASHES ====================

For details of network status enter NETSTATUS at the "Document"
prompt.
                  ------------------
Hostess catalogues revised for further details enter TOPCAT at
the "Document?" prompt.
                  ------------------
New release of VPAD software to be tested at Swindon (0793 614914).
For details type INFORMATION.VPAD at the "Document?" prompt.
                  ------------------
Changes to the Online Billing system. Please type NEWSFILE at
```

Per consultare l'elenco di tutti i NUA britannici ci siamo collegati al database Hostess

dopo qualche altro secondo il collegamento dovrebbe iniziare con il messaggio di *login* da parte del computer remoto.

L'esempio appena riportato è utilizzabile da tutti coloro che dispongono di un NUI, dal momento che *Hostess* non richiede nessun codice d'accesso. Per altre banche dati connesse in rete è invece necessario poter accedere a ulteriori codici, e questo limita un po' le possibilità di collegamento. In queste pagine troverete alcuni esempi di collegamento, resi possibili grazie alla registrazione effettuata presso le banche dati chiamate.

Chi possiede una carta di credito, può impiegarla per abbonarsi ai servizi telematici che più gli interessano, ricordando però che dovrà specificare il suo numero di telefono per poter essere poi richiamato dai gestori del servizio a conferma dei dati dichiarati. Quasi tutti i servizi americani prevedono questa possibilità.

Per evitarvi faticose ricerche, vi diamo direttamente i NUA dei servizi telematici più interessanti:

Prestel	023421920102517
PW Demo	4444444444 4444
Compuserve	0311020200202
The Source	031103010002400
BIX	0310600157878

La possibilità di collegarsi al *Prestel* inglese, ottenendo una normale visualizzazione delle pagine in formato videotex, è determinata dalla quasi assoluta trasparenza della rete a commutazione di pacchetto rispetto ai caratteri ricetrasmessi (ovviamente il vostro software di terminale andrà impostato per la visualizzazione in formato videotex). In alcuni casi però può essere necessario intervenire sul modo di funzionamento del PAD a cui siamo collegati per svolgere collegamenti che prevedono situazioni particolari, come per esempio il caricamento di file con il protocollo XMODEM.

Negli articoli che saranno dedicati nel prossimo futuro a questo tipo di reti, verranno descritti più a fondo i singoli servizi raggiungibili tramite la rete nazionale e internazionale a commutazione di pacchetto.

Quando la rete è intelligente

Non avendo la possibilità di collegarci a *ITAPAC* con una connessione dedicata e protocollo X.25, dobbiamo passare attraverso il Packet Assembler Disassembler (PAD) per poter usufruire dei servizi di rete. Il PAD non si limita a trasformare in pacchetti il flusso di dati da noi prodotto e riassemblare i pacchetti ricevuti, ma svolge anche una serie di funzioni di filtro e integrazione di quanto transita. Quando stipuliamo il contratto di utenza con la SIP dobbiamo anche richiedere che il PAD a cui siamo destinati venga programmato secondo uno dei dieci profili standard. Per il colloquio in X.28 (corrispondente al 300/300 o 1200/1200 su linea commutata) si ha generalmente l'assegnazione del profilo standard 3. Questo profilo prevede 19 parametri che possono essere comunque modificati a ogni collegamento con il PAD; se si desidera la modifica permamente di un particolare parametro possiamo fare una richiesta alla SIP che provvederà a modificarlo... e ci addebiterà L. 45.000 per l'intervento. È meglio quindi imparare come si programma un PAD per risolvere indipendentemente tutti quei piccoli problemi di connessione che possono variare a seconda del computer chiamato.

Durante un collegamento a un computer remoto siamo nella fase di "scambio dati": il PAD svolge le sue funzioni in base ai parametri che sono stati impostati, e nessun carattere viene interpretato dal PAD in maniera attiva, cioè come un comando a lui rivolto. Per poter invece accedere al PAD stesso dobbiamo entrare nel modo "comando", sempre che il profilo predefinito ce lo permetta.

Fortunatamente il profilo 3 prevede la "richiesta d'attenzione del PAD" che agisce nel caso venga inviato il carattere DLE (ottenibile premendo Control-P); questo ci permette di alterare il profilo utilizzato, modificare uno o più parametri o visualizzare l'intero stato dei parametri.

L'accesso al profilo si ottiene digitando PROF <n> con n che corrisponde al nuovo profilo. In questo modo l'utente, oltre che riprogrammare il proprio profilo, può, per tutta la durata del collegamento, scegliere uno dei profili già esistenti. Per motivi pratici il PAD torna al modo scambio dati dopo ogni comando, e quindi dovremo inviare la combinazione Control-P ogni volta che vorremo inviare al PAD un comando, anche se segue immediatamente a un altro.

Inviando il comando PAR? riceveremo dal PAD l'elenco dei 19 parametri con il loro stato, e questo può essere un buon esercizio per imparare a districarci nella tavola dei parametri e provare ad alterarli. Vediamone l'elenco completo sotto una luce non rigorosamente tecnica, ma abbastanza comprensibile.

Il parametro 1 determina la possibilità da parte dell'utente di accedere al modo comando del PAD e può assumere solo due valori: 0 (non possibile) e 1 (possibile tramite il carattere DLE).

Il parametro 2 fa sì che tutti i caratteri ricevuti dal PAD siano inviati nuovamente all'utente così come sono stati interpretati dal PAD. Anche in questo caso i valori sono 0 (niente eco) e 1 (eco).

Il parametro 3 è il primo che riguarda il controllo del flusso dei dati. In genere il PAD ha una serie di condizioni da controllare per decidere se è il caso di assemblare un pacchetto e inviarlo; questo parametro permette di programmare un particolare carattere che il PAD riconosce come segnale di "via" per l'invio del pacchetto. I valori in questo caso sono tre: 0 (nessun carattere), 2 (carattere di CR) e 126 (tutti i caratteri il cui codice ASCII è compreso tra 0 e 31, i caratteri di controllo e il carattere DEL, ASCII 127).

Il parametro 4 serve ancora per il controllo del flusso. Il PAD ha una temporizzazione interna per decidere se la pausa fra un carattere ricevuto e il successivo è tale da determinare l'invio automatico del pacchetto corrente. Questo parametro può contenere il valore 0 che indica nessuna temporizzazione, o un valore corrispondente ai ventesimi di secondo predisposti per l'attesa.

Il parametro 5 definisce la

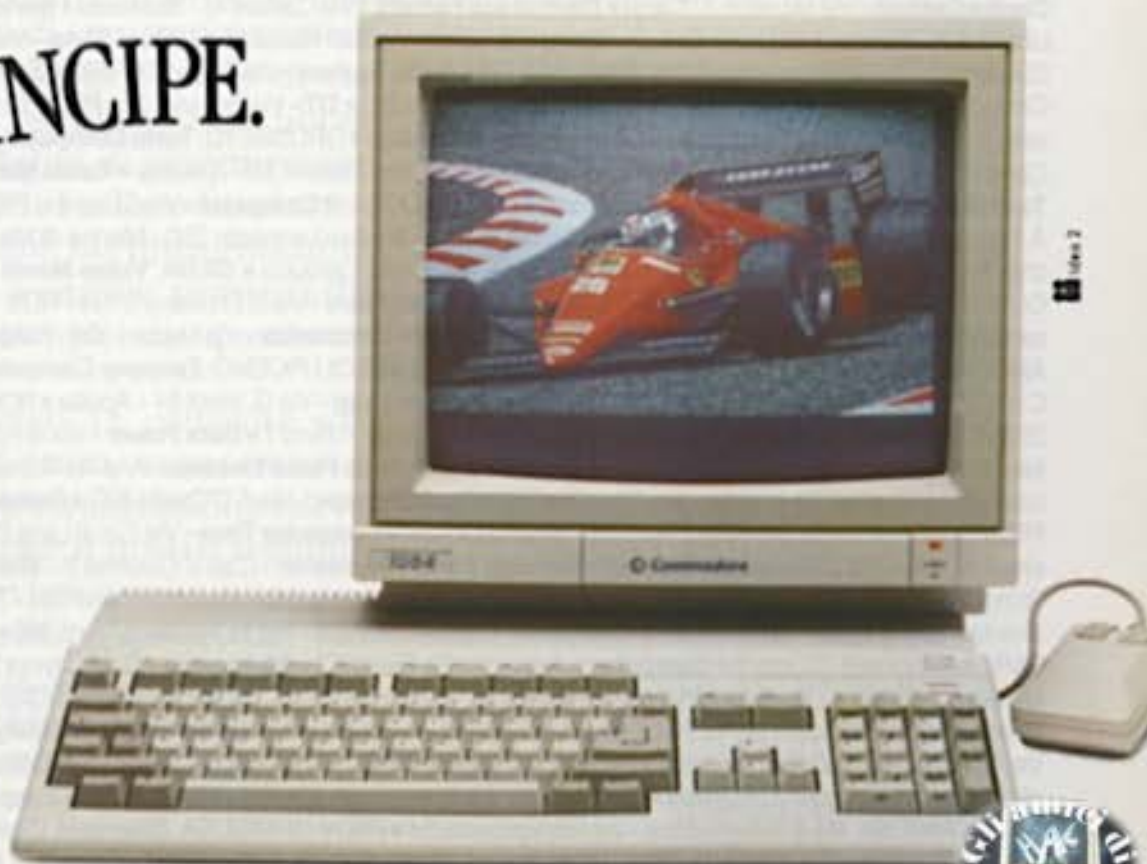
continua a pagina 53

I nuovi utenti di Compuserve ricevono questo menu per familiarizzarsi con il BBS

Amiga 500

è il principe dei computers, fiore all'occhiello della grande dinastia Commodore. Oggi puoi portarti a casa questo gioiello dell'informatica a condizioni estremamente vantaggiose: perché Commodore, se scegli un'Amiga 500, valuta il tuo vecchio computer ben **200.000** lire se è un C 64 e 100.000 lire se è un altro modello Commodore o un'altra marca.
Queste valutazioni saranno ridotte alla metà se il tuo vecchio computer non è funzionante o è incompleto.
Amiga 500 ti viene proposto in una scatola kit completa di modulatore e scrigno del software.
Corri col tuo vecchio computer nel più vicino Commodore Point o in un negozio autorizzato all'operazione "Cambia con il Principe": Amiga 500 è lì che ti aspetta.

CAMBIA IN

Commodore

200.000 LIRE

PER IL TUO VECCHIO COMPUTER

E LO CAMBIAMO CON IL PRINCIPE.

AMIGA 500. COMPUTER DELL'ANNO 1987.

ECCO I COMMODORE POINT E I PUNTI VENDITA AUTORIZZATI
CHE CAMBIANO IL TUO VECCHIO COMPUTER CON IL PRINCIPE:

Commodore

COMMODORE POINT - **LOMBARDIA** - MILANO: **Al Risparmio** - V.le Monza, 204 ▪ **Braha Alberto** - Via P. Capponi 5 ▪ **E.D.S.** - C.so Porta Ticinese 4 ▪ **E.S.C.** - Via Roggia Scagna 7 ▪ **Faref** - Via A. Volta 21 ▪ **GBC** - Via Cantoni 7 - Via Petrella 6 ▪ **Giglioni Laura** - Via G. Pacini 4 ▪ **Giglioni** - V.le L. Sturzo 45 ▪ **Logitek** - Via Golgi 60 ▪ **Marcucci** - Via F.lli Bronzetti 37 ▪ **Melchioni** - Via P. Colletta 37 ▪ **Messaggerie Musicali** - Galleria del Corso 2 ▪ **Newel** - Via Mac Mahon 75 ▪ **Rivola** - Via Vitruvio 43 ▪ PROV. DI MILANO: **F.lli Galimberti** - Via Nazionale dei Giovi 28/36 - Barlassina ▪ **GBC** - V.le Matteotti 66 - Cinisello Balsamo ▪ **P. G. Ostellari** - Via Milano 300 - Desio ▪ **Casa della Musica** - Via Indipendenza 21 - Cologno Monzese ▪ **Penati** - Via Verdi 28/30 - Corbetta ▪ **EPM** - V.le Italia 12 - Corsico ▪ **Centro Computer Pandolfi** - Via Corridoni 18 - Legnano ▪ **Computeam** - Via Vecellio 41 - Lissone ▪ **Futura** - Via Solferino 31 - Lodi ▪ **M.B.M.** - C.so Roma 112 - Lodi ▪ **L'Amico del Computer** - V.le Lombardia 17 - Melegnano ▪ **Bit 84** - Via Italia 4 - Monza ▪ **I.C.O.** - Via dei Tigli 14 - Opera ▪ BERGAMO: **Comif** - Via Autolinee 10 ▪ **Cordani** - Via dei Caniana 8 ▪ **D.R.B.** - Via Borgo Palazzo 65 ▪ **New Systems** - Via Paglia 36 ▪ PROV. DI BERGAMO: **Bertulezzi G.** - Via Fantoni 48 - Alzano Lombardo ▪ **Computer Team** - Via Verdi 1/B - Carvico ▪ **Ottico Rovetta** - P.zza Garibaldi 6 - Lovere ▪ **A.I.S. International** - Via San Carlo 25 - San Pellegrino Terme ▪ **Sisthema** - Via Roma 45 - Sarnico ▪ BRESCIA: **Computer Center** - Via Cipro 62 ▪ **Informatica 2000** - Via Stazione 16/B ▪ **Vigasio Mario** - Portici Zanardelli 3 ▪ PROV. DI BRESCIA: **Mister Bit** - Via Mazzini 70 - Breno ▪ **Cavalli Pietro** - Via 10 Giornate 14B - Castrezzato ▪ **Vietti Giuseppe** - Via Milano 1/B - Chiari ▪ **Megabyte** - P.zza Duomo 17 - Desenzano del Garda ▪ **Baresi R. & C.** - Via XX Settembre 7 - Ghedi ▪ **Info Cam** - Via Provinciale 3 - Gratacasolo ▪ COMO: **Il Computer** - Via Indipendenza 90 ▪ **2M Elettronica** - Via Sacco 3 ▪ PROV. DI COMO: **Eltrongros** - Via L. da Vinci 54 - Barzanò ▪ **EGA** - Via Mazzini 42 - Cassago Brianza ▪ **Data Found** - Via A. Volta 4 - Erba ▪ **EGA** - Via A. Moro 17 - Galbiate ▪ **Righi Elettronica** - Via G. Leopardi 26 - Olgiate Comasco ▪ **Cima Elettronica** - Via L. da Vinci 7 - Lecco ▪ **Fumagalli** - Via Cairoli 48 - Lecco ▪ CREMONA: **Mondo Computer** - Via Giuseppina 11/B ▪ **Prisma** - Via Buoso da Dovara 8 ▪ **Telco** - P.zza Marconi 2/A ▪ PROV. DI CREMONA: **Elcom** - Via IV Novembre 56/58 - Crema ▪ **Euroelettronica** - Via XX Settembre - Crema ▪ **Kofbaker** - Via Marchi 65/B - Vescovato ▪ MANTOVA: **Tubaldo E. & C.** - Galleria Fermi 7 ▪ **32 Bit** - Via C. Battisti 14 ▪ **Elettronica di Basso** - V.le Risorgimento 69 ▪ PAVIA: **Poliware** - C.so C. Alberto 76 ▪ PROV. DI PAVIA: **Logica Informatica** - V.le Monte Grappa 32 - Vigevano ▪ **M. Visentin** - C.so V. Emanuele 76 - Vigevano ▪ PIACENZA: **Computer Line** - Via G. Carducci 4 ▪ **Delta Computer** - Via Martiri della Resistenza 15/4 ▪ **Sover** - Via IV Novembre 60 ▪ SONDRIO: **Cipolla Mauro** - Via Tremogge 25 ▪ PROV. DI SONDRIO: **Fotonova** - San Pietro di Berbenno ▪ VARESE: **Dimeco Sistemi** - Via Garibaldi ▪ **Il Centro Elettronico** - Via Morazzone 2 ▪ **Supergames** - Via Carrobbio 13 ▪ PROV. DI VARESE: **Busto Bit** - Via Gavinana 17 - Busto Arsizio ▪ **Crespi G. & C.** - V.le Lombardia 59 - Castellanza ▪ **Computer Shop** - Via A. Da Brescia 2 - Gallarate ▪ **Grandi Magazzini Bossi** - Via Clerici 196 - Gerenzano ▪ **J.A.C.** - Via Matteotti 38 - Sesto Calende ▪ **PIEMONTE** - ALESSANDRIA: **Bit Micro** - Via Mazzini 102 ▪ **West Records** - C.so Roma 85 ▪ PROV. DI ALESSANDRIA: **S.G.E. Elettronica** - Via Bandello 19 - Tortona ▪ ASTI: **Record** - C.so Alfieri 166/3 ▪ CUNEO: **Rossi Computers** - C.so Nizza 42 ▪ PROV. DI CUNEO: **Punto Bit** - C.so Langhe 26/C - Alba ▪ **SDI** - Via V. Emanuele 250 - Bra ▪ **Aschieri G. Franco** - C.so E. Filiberto 6 - Fossano ▪ NOVARA: **Elcom** - C.so Mazzini 11 ▪ **Programma 3** - V.le Buonarroti 8 ▪ **Punto Video** - C.so Risorgimento 39/I ▪ PROV. DI NOVARA: **Mirco Polacco & C.** - Via Monte Zeda 4 - Arona ▪ **All Computer** - C.so Garibaldi 106 - Borgomanero ▪ **Micrologic** - Via Giovanni XXIII 2 - Domodossola ▪ **Elliott Computer** - Via Don Minzoni 32 - Intra ▪ TORINO: **Aba Elettronica** - Via C. Fossati 5/P ▪ **Alex Computer** - C.so Francia 333/4 ▪ **Computing New** - Via M. Polo 40/E ▪ **De Bug** - C.so V. Emanuele II 22 ▪ **Desme Universal** - Via San Secondo 95 ▪ **F.D.S.** - Via Borgaro 86/D ▪ **Computer Home** - Via San Donato 46/D ▪ **Informatica Italia** - C.so Re Umberto 129 ▪ **MT Informatica** - C.so G. Cesare 58 ▪ **New Business Computer** - Via Nizza 45/F ▪ **Radio TV Mirafiori** - C.so Unione Sovietica 381 ▪ **SMT Elettronica** - Via Bibiana 83/B ▪ PROV. DI TORINO: **Paul e Chico Videosound** - Via V. Emanuele 52 - Chieri ▪ **Bit Informatica** - Via V. Emanuele 154 - Ciriè ▪ **Hi-Fi Club** - C.so Francia 92/C - Collegno ▪ **I.C.S.** - Stradale Torino 73 - Ivrea ▪ **BAS** - C.so Roma 47 - Moncalieri ▪ **Cerutti Mauro** - C.so Torino 234 - Pinerolo ▪ **Eurex** - C.so Indipendenza 5 - Rivarolo C.se ▪ VERCELLI: **Elettrogamma** - C.so Bormida ▪ **Elettronica di Bellamo A. & C.** - Strada Torino 15 ▪ PROV. DI VERCELLI: **C.S.I. Teorema** - Via Losana 9 - Biella ▪ **Fotostudio Trevisan** - Via XXV Aprile 24/B - Cossato ▪ **Studio Fotografico Imarisio** - P.zza Martiri Libertà 7 - Trino.

PUNTI VENDITA AUTORIZZATI - **LIGURIA** - GENOVA: **Centro Elettronica** - Via Chiaravagna 10R ▪ **Commerciale Sottoripa** - Via Sottoripa 115R ▪ **Fotomondial** - Via del Campo 35R ▪ **La Nascente** - Via San Luca 26/28R ▪ **Play Time** - Via Gramsci 3/5/7R ▪ **Rapprel** - Via Borgoratti 23/I/R ▪ IMPERIA: **Castellino** - Via Nazionale 253 - Via Belgrano 44 ▪ PROV. DI IMPERIA: **Castellino** - Via Roma 68 - Sanremo ▪ **Centro Hi-Fi Video** - Via della Repubblica 38 - Sanremo ▪ **Castellino** - Via Genova 48 - Ventimiglia ▪ LA SPEZIA: **I.L. Elettronica** - Via Lunigiana 618 - Via V. Veneto 123 ▪ PROV. DI LA SPEZIA: **New Soft** - Via Canaletto 5 - Castelnuovo Magra ▪ **Ferrari e Lanzoni** - Via Genova 33/35 - Ceparana ▪ **I.L. Elettronica** - Via Aurelia 299 - Fornola di Vezzano ▪ SAVONA: **Castellino** - C.so Tardy e Benech 101 ▪ **LOMBARDIA** - PROV. DI MANTOVA: **Foto Anna** - Via Matteotti 51 - Poggiorusco ▪ **VENETO** - BELLUNO: **Up to Date** - Via V. Veneto 43 ▪ PADOVA: **Bit Shop** - Via Cairoli ▪ **Computer Point** - Via Roma 63 ▪ **G. Franco Marcato** - Via Madonna della Salute 51/53 ▪ PROV. DI PADOVA: **Compumania** - Riviera Tiso 37 - Camposanpiero ▪ PROV. DI TREVISO: **Ires** - Via Dante 1 - Cessalto ▪ **De Marin Computers** - Via Matteotti 142 - Conegliano ▪ **Sidestreet** - Via S. D'Acquisto 8 - Montebelluna ▪ VENEZIA: **Caputo Ruggero** - P.zza San Marco 5193 ▪ PROV. DI VENEZIA: **Ghegin** - Via Miranese 283 - Chirignago ▪ **Guerra E. & C.** - Via Bissuola 20/A - Mestre ▪ **Paccinotti** - Via Caneve 94 - Mestre ▪ VERONA: **Cartolibreria Fiscale** - Via del Pontiere 24 ▪ **Casa della Radio** - Via Cairoli 10 ▪ **Personal Ware** - Vicolo Volto San Luca ▪ PROV. DI VERONA: **Castagnetti** - Via Strà 19 - Caldiero ▪ **Bussola Luciano** - Via Brennero 54 - Domegliara ▪ PROV. DI VICENZA: **Marangoni Giuseppe** - Via Marconi 8 - Schio ▪ **FRIULI VENEZIA GIULIA** - TRIESTE: **Avanzo Giacomo** - P.zza Cavana 7 - C.so Italia 17 ▪ **Computer Shop** - Via P. Reti 6 ▪ **Universal Tecnica** - C.so Saba 18 ▪ **EMILIA ROMAGNA** - BOLOGNA: **Computer Facile** - Via Don Minzoni 4/B ▪ **Minnella Alta Fedeltà** - Via Mazzini 146/2 ▪ **Morini & Federici** - Via Marconi 28/C ▪ PROV. DI BOLOGNA: **S.P.E. Informatica** - Via di Mezzo Ponente 383/A - Crevalcore ▪ **Archimede Sistemi** - Via Emilia 124 - San Lazzaro di Savena ▪ FERRARA: **Soft-Gallery** - Via Mortara 30 ▪ FORLÌ: **Computer Video Center** - Via Campo di Marte 122 ▪ **Ellegi Computer** - C.so Garibaldi 87 ▪ PROV. DI FORLÌ: **Top Bit** - Via Veneto 12 - Forlimpopoli ▪ **Computer House** - V.le Tripoli 193/d - Rimini ▪ MODENA: **Centro Calcolo** - Via Muzzioli 18 ▪ **Orsa Maggiore** - P.zza Matteotti 20 ▪ PROV. DI MODENA: **Centro Calcolo** - Via Muratori 3 - Carpi ▪ **Elettronica Ferretti** - Via Cialdini 41 - Sassuolo ▪ **Bertoni B. Machines** - C.so Italia 28 - Vignola ▪ PROV. DI RAVENNA: **Argnani** - P.zza Libertà 5/A - Faenza ▪ REGGIO EMILIA: **Computer Line** - Via San Rocco 10/C ▪ **Pool Shop** - Via Emilia S. Stefano 9/C ▪ PROV. DI REGGIO EMILIA: **Macchioni Armando** - Via Statale 467,27 - Casalgrande ▪ **TOSCANA** - AREZZO: **Delta System** - Via Piave 13 ▪ FIRENZE: **Atema** - Via B. Marcello 1/A ▪ **Caff** - Via Allori 52 ▪ **Elettronica Cento Stelle** - Via Cento Stelle 5/A ▪ **New Computer Service** - Via degli Alfani 2/R ▪ **TIT** - Via Bronzini 36 ▪ PROV. DI FIRENZE: **War Games** - Via R. Sanzio 126 - Empoli ▪ **New EVM** - Via degli Innocenti 2 - Figline Valdarno ▪ **Atema** - Via Pisana 407 - Scandicci ▪ GROSSETO: **Tutto Computer** - Via Gramsci 2/A ▪ LIVORNO: **Eta Beta** - Via San Francesco 30 ▪ **Futura 2** - Via Cambini 19 ▪ PROV. DI LIVORNO: **Bonanni Elettronica** - Via Rosmini 3/A - Cecina ▪ **Radio Mechi** - C.so Matteotti 124 - Cecina ▪ **Tesi Adriano** - Via Carducci 4 - Piombino ▪ **Tomi Gusmano** - Via Petrarca 109 - Piombino ▪ PROV. DI LUCCA: **Il Computer** - V.le Colombo 216 - Lido di Camaiore ▪ **Pellegrini 1** - Via Oberdan 65 - Pietrasanta ▪ **Logica** - Via A. Fratti 165 - Viareggio ▪ PROV. DI MASSA: **Giorgi G. & C.** - Via San Leonardo 350 - Marina di Massa ▪ PISA: **C.H.S.** - Via C. Cattaneo 90/92 ▪ **Electronic Service** - Via della Vecchia Tramvia 10 ▪ **IT-LAB** - Via Marche 8/A/B ▪ **Tony Hi-Fi** - Via Carducci ▪ SIENA: **Video Movie** - Via Garibaldi 17 ▪ PROV. DI SIENA: **Bifolchi Giordano** - Via di Gracciano nel Corso 111 - Montepulciano ▪ **UMBRIA** - PERUGIA: **Migliorati Piero** - Via S. Ercolano 3/10 ▪ PROV. DI PERUGIA: **Computer Studios** - Via IV Novembre 18/A - Bastia Umbra ▪ **Tonzani Orietta** - Via G. Di Vittorio 13/A/B/C - Ellera ▪ **Marinelli Elettronica** - Via Mazzini 104 - Foligno ▪ TERNI: **Ramozzi Rossana** - Via Porta S. Angelo 23 ▪ **MARCHE: PROV. DI ANCONA: Bit e Video** - C.so Matteotti 28 - Jesi ▪ PROV. DI ASCOLI PICENO: **Zerouno Computer** - Via Voltatorni - San Benedetto del Tronto ▪ MACERATA: **Giannobi Muzio** - C.so Cavour 93 ▪ **LAZIO** - PROV. DI LATINA: **Marcheggiani Luigi** - Via G. Verdi 64 - Aprilia ▪ ROMA: **Big Byte** - Via V.G. De Vecchi Pieralice 35 ▪ **Compushop** - Via Nomentana 265 ▪ **Computer Friend** - Via A. Romano 3 ▪ **Computron** - L.go Forano 7 ▪ **Data Power** - Via di Fontana Candida 2/C ▪ **Due Emme Elettronica** - Via Britannia 17 ▪ **Egis Computer** - Via Castro dei Volsci 42 ▪ **Elettromarket** - Via C. Balbo 1 ▪ **La Placa Vincenzo** - Via Val Trompia 12/18 ▪ **Leonardo** - Via Chopin 29 - Via P. Castaldi 7 ▪ **Metro Import** - Via Donatello 37 ▪ **Musical Cherubini** - Via Tiburtina 360 ▪ **Pix Computer** - Via F. D'Ovidio 6/C ▪ **Romana Componenti Elettronici** - P.zza dei Gerani 40/41 ▪ **R.P.M.** - Via Giulia 142 ▪ PROV. DI ROMA: **Delta Bit** - Via G. Verdi 26 - Albano Laziale ▪ **Computer Time** - Via Col di Lana 11/15 - Ciampino ▪ **L'Angolo del Computer** - Via delle Case Nuove 3 - Civitavecchia ▪ **M.R.S.** - Via L. Manara 11 - Frascati ▪ **Marzetti Elettrodomestici** - C.so V. Colonna 11 - Marino ▪ **Bit House** - Via Kennedy 100 - Monterotondo ▪ **Lab. P. D'Alberti** - V.le dei Promontori 148 - Ostia Lido ▪ **Paolini** - Via Paolini 94 - Ostia Lido ▪ **A.V.C.** - Via Empolitana 134 - Tivoli ▪ **Computer Shop** - Via 2 Giugno 34 - Tivoli ▪ VITERBO: **Treac** - Via Palazzina 1 ▪ **ABRUZZO** - PESCARA: **Chip Computer** - Via Milano 77/8 - Via N. Adriatica Nord 386 ▪ **BASILICATA** - MATERA: **G. Gaudiano Electronics** - Via Roma 2 ▪ **PUGLIA** - BARI: **Artel** - Via G. D'Orso 9 ▪ **Computer's Arts** - Via Re David 171 ▪ **Discorama** - C.so Cavour 99 ▪ PROV. DI BARI: **Zingaro e Zagaria** - Via Torino 26/28 - Andria ▪ **Faggella Gianni** - Via Alvisi 4 - Barletta ▪ **Pietrantonio G. & C.** - Via Vavalle 2/A - Conversano ▪ BRINDISI: **Olivieri** - Via Bezzecca 9 ▪ **FOGGIA: Botticelli Guido** - Via Sav. Pollice 2 ▪ **La Torre** - V.le Michelangelo 185 ▪ LECCE: **Bit** - Via 95° Reg. Fanteria 87/89 ▪ TARANTO: **Elettrojolly** - Via De Cesare 13 ▪ **T E A** - Via Regina Elena 101 ▪ **SICILIA** - CATANIA: **A Zeta** - Via Canfora 140 ▪ **C.D.M.P.** - Via Amantea 4 ▪ **Elettronica Delta** - Via Messina 413 B ▪ **Foto Ottica Randazzo** - L.go dei Vespri 21 ▪ **Paratore** - Via Maddem 141 ▪ MESSINA: **Mister Bit** - Via Nazionale 10 ▪ **Office Automation** - Via G. Venezian 75 ▪ PALERMO: **Home Computers** - V.le delle Alpi 50/F ▪ **Randazzo Angelo** - Via R. Settimo 53/55 - [illegible] 20/40 ▪ RAGUSA: **Medi Informatica** - Via Risorgimento 54 ▪ PROV. DI RAGUSA: **Giannone Computers** - Via Vanella Macallè - Modica ▪ SIRACUSA: **Computer S**[illegible] - Via San Simeone 15 ▪ PROV. DI SIRACUSA: **Amore Maria Inserra & C.** - Via Garibaldi 15 - Lentini.

predisposizione del PAD a inviare i codici di X-ON e X-OFF. Se il valore del parametro è 0 il PAD filtra le richieste d'interruzione e riavvio del flusso di dati inviate dal nostro terminale. Se il valore è invece 1, il PAD interrompe la nostra trasmissione con X-ON e la riattiva con X-OFF per adattarla alle necessità del computer a cui siamo collegati.

Il parametro 6 controlla i segnali di servizio inviati dal PAD al nostro terminale. I possibili valori sono tre: 0 (nessun segnale), 1 (solo i segnali di servizio), 2 (anche il prompt).

Il parametro 7 definisce l'azione del PAD al ricevimento di un "break" inviato dal nostro terminale. Se il parametro vale 0 non verrà compiuta nessuna azione. Se vale 1 reagirà con un interrupt; se vale 2 si otterrà un reset dei parametri a seconda della configurazione del profilo a noi assegnato; se vale 8 il PAD uscirà dal modo dati per entrare nel modo comandi, e infine se il parametro è programmato con il valore 21 si otterrà lo scarto del pacchetto in fase di trasmissione verso di noi, un interrupt e l'indicazione del ricevimento di un "break".

Il parametro 8 agisce sullo scarto dei dati in uscita dal PAD verso il nostro terminale. Se questo parametro è a 0 ci perverranno tutti i dati a noi destinati; se invece è a 1 non riceveremo nessun dato dal computer remoto a cui siamo collegati. Per ripristinare la ricezione possiamo sempre impostare a 0 il parametro, anche durante il collegamento, con il comando Control-P RESET o con il comando Control-P SET8:0.

Il parametro 9 definisce la lunghezza del ritardo da inserire in fase di trasferimento dati dal PAD al nostro terminale dopo un Carriage Return. Questo comando va visto nell'ottica di una stampante connessa alla rete (una telescrivente, ad esempio) che necessita di un certo tempo per riportare il carrello all'inizio della riga. Con questo parametro possiamo far sì che il PAD invii alcuni caratteri di riempimento il cui numero può variare da 0 a 7.

Il parametro 10 è sempre dedicato ai terminali di tipo telescrivente, e permette di programmare il numero di caratteri trasmessi dal PAD prima dell'inizio di ciascuna riga. In questo caso abbiamo la possibilità di programmarne da 0 a 255. La loro funzione è quella di dare un'impaginazione di base (un margine sinistro) durante la stampa di quanto si riceve.

Il parametro 11 consente solo accessi in lettura e contiene la velocità di ricetrasmissione rilevata dal PAD per il nostro terminale. A prima vista questa frase può sembrare incongruente, dal momento che la velocità di trasmissione dovrebbe essere quella opportunamente selezionata (300 o 1200 baud). Ma se riflettiamo, ci accorgiamo che durante un collegamento esistono molti tempi morti durante i quali, pur essendo il collegamento a 300 baud – o la cifra che preferite – non viene trasmesso neanche un bit, e quindi l'effettiva velocità di trasferimento dei dati diminuisce drasticamente, fino a diventare zero durante molti periodi di trasmissione.

X.28 prevede la trasmissione asincrona e quindi, esaminando i primi due Carriage Return ricevuti, il PAD è in grado di determinare la nostra effettiva velocità di ricetrasmissione dati. Le possibili velocità di ricetrasmissione sono:

0 - 110 bit/s,
2 - 300 bit/s,
8 - 200 bit/s,
3 - 1200 bit/s,
4 - 600 bit/s,
6 - 150 bit/s,
10 - 50 bit/s.

Ecco all'interno del BBS BIX un esempio di dibattito telematico tra gli utenti

Il parametro 12 è simile al parametro 5, solo che in questo caso è l'utente a gestire il flusso di dati con il PAD, tramite i caratteri di X-ON e X-OFF. Se il parametro è programmato a 0 il PAD ignora le nostre richieste, se invece è a 1 potremo utilizzare questi caratteri speciali per gestire il flusso dei dati in arrivo, interrompendolo e riattivandolo a piacere.

Il parametro 13 determina la presenza del codice di Line Feed dopo un Carriage Return... quante volte non abbiamo litigato con una stampante che stampa tutto su una sola riga o stampa a righe alternate? Per evitare questi problemi, non solo con una stampante ma anche con il video, questo parametro prevede tutte le possibili combinazioni di Line Feed e Carriage Return verso il nostro terminale e verso il computer remoto. Ripetiamo qui i possibili valori che questo parametro

può assumere:
0 - nessun inserimento di un LF
1 - inserimento di un LF dopo ogni CR nei dati verso l'utente
4 - inserimento di un LF dopo ogni CR in eco dal PAD all'utente
5 - inserimento di un LF dopo ogni CR sia in eco sia nei dati verso l'utente
6 - inserimento di un LF dopo ogni CR dal terminale dell'utente e dopo un CR in eco al terminale dell'utente
7 - inserimento di un LF dopo ogni CR da e verso il terminale dell'utente, e dopo ogni CR in eco verso il terminale dell'utente.

BIX consente ai suoi utenti di accedere anche a tutti gli articoli pubblicati da Byte

Il parametro 14 viene utilizzato anch'esso per dare una mano a quei terminali che eseguono la stampa diretta di quanto ricevono. Come per il parametro 9 è possibile far sì che il PAD invii alcuni caratteri appena dopo il carattere Line Feed, dando il tempo alla stampante di provvedere all'avanzamento della carta. Il valore minimo è 0 e il massimo è 7.

Il parametro 15 è invece importante per tutti i collegamenti non automatizzati. Dato che ogni segmento (64 byte) ha un costo, può essere utile rivedere il pacchetto di dati che si sta assemblando nel PAD per effettuare delle correzioni. Se il parametro è a 0 non possiamo fare nessuna correzione, se è a 1 possiamo invece utilizzare i comandi di edit (spostamento del cursore...) per intervenire, prima d'inoltrarla in rete, sull'ultima stringa che abbiamo digitato.

Il parametro 16 riguarda ancora l'edit e serve per definire quale carattere il PAD deve interpretare come Delete. Questa flessibilità consente di compensare le eventuali limitazioni del nostro terminale. Il valore standard è 127 che corrisponde al codice ASCII DEL, ma se il programma di comunicazione, o il computer stesso, non prevedono questo carattere, si può riprogrammare il parametro per accettare un qualsiasi carattere che abbia un valore compreso fra 0 e 127.

Il parametro 17 definisce il carattere di cancellazione della riga. Il valore standard è 24, corrispondente a Control-X. Anche per questo parametro è possibile modificare il valore imponendone uno compreso fra 0 e 127.

Il parametro 18 è di notevole utilità per tutte le situazioni in cui non si riceve l'eco dei caratteri che vengono digitati. Può predisporre la visualizzazione del pacchetto in fase di assemblaggio nel PAD, in seguito all'invio di un particolare carattere. Il valore standard è 18 (Control-R) e può essere modificato con un valore compreso fra 0 e 127.

Il parametro 19 è l'ultimo ed è abbastanza recente. Permette il controllo dei segnali di servizio per l'edit. Ancora una volta il PAD si preoccupa di quale tipo di terminale abbiamo a disposizione e predispone i segnali di controllo per un terminale video o per una stampante. Se è a 0 non abilita i segnali di servizio, a 1 li abilita per la stampante e a 2 li abilita per il video.

Con tutti questi parametri è davvero possibile personalizzare la rete secondo le più svariate esigenze; è comunque utile conoscere il valore dei parametri corrispondenti al profilo 3, assegnato per default ai terminali che si collegano tramite la linea commutata.

I valori dei parametri sono: 1=1, 2=1, 3=126, 4=255, 5=1, 6=5, 7=21, 8=0, 9=2, 10=80, 11=attivo, 12=1, 13=5, 14=1, 15=0, 16=127, 17=24, 18=18 e 19=2.

Un'ultima comodità offerta dai nodi è la presenza di due servizi speciali chiamati ECO e DROP, atti a verificare l'esatto funzionamento dei propri collegamenti. Si attivano digitando un NUA particolare e permettono di controllare l'interpretazione da parte del PAD di quello che viene digitato. ECO fa sì che il PAD invii nuovamente al terminale ogni pacchetto che ha ricevuto dal terminale, simulandone quindi un ipotetico "viaggio" nella rete, mentre DROP si limita a riscontrare la corretta applicazione del protocollo di trasmissione e a documentarla all'utente.

I NUA per questi due servizi sulla rete sono i seguenti:

	ECO	DROP
Milano	22000	22001
Torino	21100	21101
Firenze	25500	25501
Roma	26000	26001
Napoli	28100	28101

Ora manca solo una breve spiegazione dei messaggi di...

possono ricevere dalla rete durante i collegamenti, dopodiché *ITAPAC* non avrà più alcun segreto. I messaggi possono essere solo di due tipi: di non-collegamento e di PAD.

I messaggi del primo tipo spiegano i motivi della mancata connessione o dell'interruzione del collegamento con un computer remoto e sono i seguenti:

CLR DTE
il collegamento è stato interrotto dal computer remoto

CLR CONF
il PAD ha interrotto il collegamento su richiesta dell'utente (avete chiamato il PAD con Control-P e gli avete inviato il comando CLR)

CLR OCC
tutte le porte dalla rete al computer chiamato sono occupate

CLR NC
la rete è congestionata e non vi permette il collegamento

CLR INV
avete richiesto una funzione non valida

CLR NA
l'accesso al computer richiesto non è permesso

CLR ERR
avete inviato una richiesta errata

CLR RPE
il computer chiamato ha commesso un errore di procedura

CLR NP
il NUA chiamato non è disponibile

CLR DER
il NUA chiamato è guasto

CLR PAD
il computer chiamato ha ordinato al PAD di scollegarci

CLR NRC
abbiamo provato a chiamare a carico del destinatario (aggiungendo una R al termine del NUI) un computer che non lo permette

I messaggi di PAD sono invece indicativi di qualche evento che non interrompe il collegamento ma determina la perdita del pacchetto corrente.

RESET DTE
Reset del PAD su richiesta del computer remoto

RESET ERR
Reset causato da un errore di procedura

RESET NC
Reset causato da una congestione della rete

ERROR
comando errato

INV
parametro o comando errato

ILL
richiesta non permessa (in genere appare quando si digita un NUI errato) ■

GLOSSARIO

ACP Adattatore Concentratore di Pacchetto, racchiude le funzioni di PAD, d'interfaccia verso i DTE e concentra i pacchetti per l'invio ai nodi.

DCE Data Circuit Termination Equipment, il dispositivo utilizzato per il collegamento al mezzo trasmittente. Ad esempio un modem è indicato come DCE.

DNIC Data Network International Code, il "prefisso" delle reti internazionali.

DTE Data Terminal Equipment, terminale dati.

IA5 International Alphabet 5, altro termine usato per indicare il set ASCII a 7 bit.

ISDN Integrated Services Digital Networking, rete digitale che integra fra loro tutti i possibili dispositivi di comunicazione garantendo l'interconnessione a velocità estremamente elevata.

NCP Nodo a Commutazione di Pacchetto, elaboratore in grado di gestire il flusso dei dati all'interno della maglia della rete a cui è connesso.

NUA Network User Address, l'equivalente del numero telefonico per la rete commutata.

NUI Network User Identifier, il codice d'identificazione dell'utente verso la rete stessa.

PAD Packet Assembler Disassembler, l'interfaccia messa a disposizione dalla rete per trasformare il nostro colloquio X.28 in pacchetti X.25.

Porta Risorsa di accesso alla rete.

PSS Packet Swich Stream, rete a commutazione di pacchetto.

SWIFT Society for Worldwide Interbank Financial Telecommunications, la rete internazionale che collega tutte le banche per le trasmissioni di transazioni valutarie.

X.25 Raccomandazioni per le specifiche di comunicazione nelle reti a commutazione di pacchetto.

X.28 Raccomandazioni per le specifiche di comunicazione in modalità start/stop.

COME CREARSI UN VIDEOGIOCO "FATTO IN CASA"

Prendiamo in esame due noti generatori di adventure – The Graphic Adventure Creator e Adventure Construction Set – e un generatore di arcade: Shoot'em up Construction Kit. Caratteristiche, limiti e consigli utili

di Franco Toldi

La pubblicazione da parte della Electronic Arts di *Adventure Construction Set*, un programma che mette gli utenti dell'Amiga in grado di creare con grande facilità giochi di ruolo, ci offre l'occasione di fare il punto su questo particolare tipo di tool.

La via era stata aperta da strumenti come *The Quill* della Gilsof, indirizzato alla creazione di giochi d'avventura con solo testo, e come *Game Maker* dell'Activision, un ottimo generatore per giochi di risalita e simili. Ora con gli ultimi arrivati, *The Graphic Adventure Creator* della Incentive Software e *Shoot'em up Construction Kit* della Outlaw, gli aspiranti autori hanno a disposizione per il C-64 raffinati strumenti a livello semiprofessionale.

Strumenti di questo livello non erano ancora disponibili per l'Amiga. *Adventure Construction Set*, pur con i limiti che più avanti vedremo, apre anche per questa macchina l'era dei generatori. Ci auguriamo che, come è accaduto per il C-64, siano presto disponibili altri programmi che permettano di accedere alle notevoli doti grafiche e musicali dell'Amiga, sollecitando così la creatività dei suoi utenti.

In questo articolo esaminiamo i tre programmi, lasciando per ultimo quello destinato all'Amiga, e mettendo in evidenza i loro pregi e i loro difetti. Una particolare attenzione è riservata a consigli per servirsi di questi programmi e diventare così veri e propri "creatori di adventure".

The Graphic Adventure Creator

Prodotto in diverse versioni ad uso degli utenti dell'Amstrad, dello Spectrum e infine del C-64, questo programma rappresenta la punta più avanzata in materia di tool dedicati alla realizzazione di giochi d'avventura.

Una grafica di semplice impiego ma utilizzabile per realizzazioni complesse, un parser capace di analizzare più ordini in sequenza e un conciso ma completo manuale, fanno di questo programma un ottimo strumento sia per il neo-autore di avventure, sia per coloro che pensano a uno sviluppo semiprofessionale della loro creatività.

La preparazione

Questo pacchetto, come altri dello stesso tipo, presuppone che l'utente abbia preventivamente definito i parametri base della sua avventura. In particolare, che abbia preparato una traccia della storia, deciso in quante locazioni si articolerà, steso sommariamente una descrizione delle locazioni stesse, tracciato una mappa del gioco e previsto gli input del giocatore. Meglio ancora se ha tracciato uno schizzo delle diverse illustrazioni che accompagneranno il gioco e ha già scomposto gli input nelle diverse categorie grammaticali utilizzate dal generatore. Si tratterà naturalmente solo di un abbozzo, in quanto il generatore consente ampi interventi in fase di correzione, ma il lavoro successivo ne sarà molto facilitato.

Il menu principale

Il menu principale elenca le opzioni disponibili per il nostro lavoro in ordine alfabetico. L'ordine con cui si impiegano queste opzioni nella stesura di un gioco è naturalmente frutto di scelte personali e può benissimo essere diverso da quello proposto

queste pagine. Il nostro è un esempio che può essere utile come indicazione di base.

Per cominciare, esaminiamo le diverse opzioni del generatore, con una breve spiegazione dello scopo di ognuna. Attraverseremo così, passo dopo passo, le diverse fasi della costruzione di un'ipotetica avventura.

VERBS, i verbi

Per il sistema, i "verbi" sono le parole con cui il giocatore comunica al computer ciò che il suo personaggio deve fare. Il sistema consente di costruire un vocabolario ricco al massimo di 255 verbi. Trattandosi dell'elemento linguistico di maggiore rilevanza nella comunicazione giocatore/computer, sarà buona norma far seguire a ciascun verbo un adeguato numero di sinonimi.

I comandi di direzione come SUD, NORD e così via, vengono trattati dal sistema come verbi. Anche in questo caso sarà bene prevedere sia la versione estesa sia quella abbreviata.

Per l'introduzione dei "verbi" nel vocabolario bastano alcune semplici operazioni da tastiera chiaramente descritte nell'indispensabile manuale.

ROOM DESCRIPTION, le stanze

Come "stanze" o "locazioni" il sistema intende i diversi luoghi in cui vogliamo ambientare il nostro gioco. Le istruzioni ci garantiscono l'impiego di ben 9999 locazioni. Naturalmente sarà decisiva non solo la nostra fantasia, ma l'impiego di memoria che comporterà la descrizione di ciascuna. A disposizione per ogni descrizione abbiamo infatti 255 caratteri.

Iniziamo il nostro lavoro dal menu principale premendo il tasto R. Comparirà quindi uno schermo con la richiesta di sistema del numero della stanza, la comunicazione «Room number n is...?». Nella stessa schermata potremo introdurre la descrizione da noi preparata.

A questo punto dobbiamo precisare le connessioni tra questo ambiente e gli altri. Alla richiesta «Connections are...?», faremo seguire uno dei verbi del nostro vocabolario, seguito da uno spazio e dal numero relativo alla stanza con cui desideriamo stabilire una comunicazione. Scriveremo, ad esempio, «Ovest 11, Nord 7». Una stanza può ovviamente avere più di una porta, e condurre a più locazioni: basta elencare i relativi verbi e i numeri distintivi.

Il sistema a questo punto ci richiede d'indicare il numero dell'immagine che desideriamo compaia nella schermata relativa a questa stanza. Se abbiamo già provveduto alla parte grafica dell'avventura, non dobbiamo far altro che fornire il numero dell'immagine. In caso contrario premeremo Return. Una volta preparate le immagini, torneremo alle stanze facendo le opportune correzioni.

MESSAGES, le comunicazioni

Con questa opzione il generatore elabora tutti i messaggi che noi desideriamo far pervenire al giocatore, sia di tipo narrativo sia di sistema. Ci è consentito immagazzinare sino a 255 diversi messaggi, ciascuno composto da 255 caratteri.

Le operazioni si svolgono seguendo una procedura analoga a quella sopra accennata e non pongono problemi di sorta. Particolare attenzione dobbiamo dedicare ai messaggi che vanno dal numero 240 al numero 255. Si tratta di messaggi di sistema che il generatore sa già come interpretare e di cui necessita per il funzionamento. Il loro elenco ci viene fornito dall'appendice A del manuale. Non ci resta, ove lo ritenessimo opportuno, che tradurli o modificarli, mantenendo comunque il loro significato.

You find yourself on the bank of a turbulent stream babbling along the base of the mountain itself. It is crossed by a stone bridge leading East to a dark cave entrance.You can also see an old oil lamp
What now?...

Un'avventura dimostrativa creata interamente con The Graphic Adventure Creator

NOUNS, OBJECTS, nomi, oggetti

Il mondo delle avventure contiene – accanto a luoghi meravigliosi e personaggi fantastici – anche innumerevoli oggetti dalle funzioni più diverse. Il generatore vede questi oggetti sotto due aspetti. Da un lato sotto l'aspetto della comunicazione, e prevede quindi un elenco di "nomi". Dall'altro sotto l'aspetto del loro impiego (facilità di trasporto, manipolazione...) e prevede quindi un elenco di "oggetti".

Per il primo elenco seguiremo

una procedura simile a quella richiesta per i verbi. Per ogni oggetto, oltre a indicare il numero distintivo daremo (seguendo le comunicazioni di sistema) una descrizione lunga sino a 255 caratteri e indicheremo la "stanza" dove dovrà essere riposto.

ADVERBS, avverbi

Con questa opzione, il generatore tratta parole atte a precisare il significato sia di verbi sia di nomi. Permette così di differenziare "corro velocemente" da "corro", oppure "scatola nera" da "scatola rossa" o da "scatola".

Per il loro inserimento seguiremo le istruzioni previste, che richiedono una procedura simile a quella dei verbi.

CONDITIONS, condizioni

Con questa opzione il generatore permette di stabilire le alternative che simuleranno un processo decisionale nel gioco. I vari elementi sin qui esaminati, verbi, nomi, avverbi, oggetti, stanze e messaggi devono ora essere integrati nelle costruzioni "condizionali" che formano la struttura del gioco.

Possiamo pensare alle "conditions" come a ordini che diamo alla macchina affinché compia determinate operazioni, qualora siano verificate opportune condizioni. Analogamente al costrutto IF... THEN..., costruiamo per il nostro gioco ben più complesse condizioni. Così, ad esempio, se assegniamo al verbo "esamina" il numero 7 e al nome "cucina" il numero 11, possiamo scrivere: IF (VERB 7 AND NOUN 11) LOOK WAIT END. Così, nel gioco, se l'input sarà «Esamino la cucina», il sistema provvederà a far comparire la descrizione del locale e quindi resterà in attesa di un nuovo messaggio del giocatore.

Nel manuale che accompagna il generatore sono ampiamente elencati con molti esempi gli elementi costitutivi di questo facile "linguaggio". Con il generatore è altresì fornita una piccola avventura nella modalità di file, accessibile – tramite l'opzione "test" – al nostro esame e alle nostre prime esercitazioni. Tornando al generatore, questo, tramite opzioni separate, prevede tre diversi tipi di "conditions".

High Conditions. Sono le condizioni che vogliamo siano controllate dal sistema prima che il giocatore introduca il suo input. Così, ad esempio, se abbiamo assegnato al personaggio un certo numero di "vite", quando sono terminate il sistema dovrà informare di ciò il giocatore e far apparire la scritta finale.

Local conditions. Il sistema del gioco controlla queste condizioni subito dopo l'introduzione dell'input da parte del giocatore. Si tratta di condizioni che hanno a che vedere con il luogo specifico in cui il personaggio si trova. Così, ad esempio, se il personaggio si trova in una radura e ne chiede la descrizione, il sistema provvederà a farlo restando quindi in attesa del successivo input.

Low priority conditions. Come le precedenti, sono controllate dal sistema dopo l'input del giocatore. Esse non hanno tuttavia a che vedere con la specifica stanza in cui il personaggio si trova, non essendo associate con nessun luogo in particolare. Così se in una stanza qualsiasi il giocatore chiede l'inventario degli oggetti che porta con sé, il sistema provvederà alla stampa di questa informazione.

Il parser

I giochi d'avventura costruiti con questo generatore accettano dal giocatore input di una certa complessità, che possono essere formati da uno o più comandi della forma "verbo, avverbio, nome" o della forma "verbo, nome, avverbio" legati tra loro dai diversi segni d'interpunzione. Tutte le lettere che formano le parole comprese nel vocabolario sono significative. A differenza di altri generatori, non c'è il troncamento della parola esaminata alla quarta o quinta lettera. Ricordiamo che possiamo introdurre le abbreviazioni desiderate facendo ricorso allo stesso metodo che abbiamo spiegato precedentemente per i sinonimi.

La grafica

Questo generatore attribuisce all'elemento grafico la massima importanza. Mette quindi a disposizione uno strumento in grado di fornire risultati quantomeno accettabili. Tramite l'opzione "Graphics" del menu principale accediamo a uno schermo in cui è aperta un'ampia finestra ad alta definizione corredata da un menu grafico. I comandi a disposizione sono limitati all'essenziale ma non escludono opzioni di una certa raffinatezza. Per il comando Fill, ad esempio, è disponibile un'opzione che ne prevede l'uso in modalità "pattern". Così con il "Picture merge" è possibile unire il disegno su cui stiamo lavorando con un altro precedentemente archiviato.

Altre opzioni

Il generatore qui esaminato è dotato di altre opzioni destinate a facilitare il nostro lavoro. Per esempio abbiamo la possibilità di salvare quanto è stato realizzato sia sotto forma di "file" sia come "runnable adventure". Possiamo quindi suddividere il nostro lavoro in diverse fasi successive, certi di poterlo riprendere senza difficoltà.

L'opzione "Test" ci consente infine di provare la nostra costruzione prima del salvataggio definitivo nella forma "runnable adventure", non più modificabile da generatore.

Osservazioni

The Graphic Adventure Creator è un generatore completo. Nonostante la sua semplicità, si colloca senza dubbio nell'area semiprofessionale.

Nel mondo anglosassone ha incontrato un notevole successo, al punto che alcune riviste del settore (come A.C.E.), gli dedicano rubriche periodiche. Giochi realizzati con questo generatore vengono commercializzati in Inghilterra con buoni risultati nella categoria del budget software.

Shoot'em up Construction Kit

Tempo fa, *Game Maker* della Activision aveva messo a disposizione degli aspiranti autori di "arcade" uno strumento senza dubbio valido, ma che comportava tali e tante limitazioni da incidere sul risultato globale. In particolare, l'impossibilità di programmare lo scorrimento dello schermo limitava il campo ai soli giochi di risalita o simili, escludendo totalmente gli Shoot'em up.

Il generatore di cui parleremo ora ha colmato questa lacuna fornendo ai futuri autori uno strumento ottimale. Chiunque infatti, senza alcuna conoscenza di linguaggio macchina, Basic o altro, può facilmente creare giochi ben rifiniti e perfettamente funzionanti. Il tutto muovendosi agilmente tra menu e sotto-menu tramite joystick. Questo generatore non è soltanto una raccolta di utility. È un tool specializzato per la costruzione di giochi d'azione, completo in ogni sua parte, dagli effetti sonori alla presentazione.

Shoot'em up Construction Kit consente la creazione di giochi anche molto sofisticati

Nell'esaminare questo programma ci lasceremo guidare dalla struttura stessa del tool. D'altra parte, come è stato chiarito dagli stessi autori in una recente intervista, è proprio questo il metodo che loro suggeriscono. In caso di reale impiego del tool, e non solo di una seduta di apprendimento, ognuno procederà ovviamente secondo le proprie abitudini di lavoro. Il programma garantisce infatti ampia libertà al riguardo, fornendo le necessarie opzioni di salvataggio parziale del lavoro fatto.

MAIN MENU, il menu principale

Una volta lanciato il programma, sullo schermo compare un menu a cinque scelte. Quattro fanno capo a dimostrazioni realizzate con il generatore. Uno al generatore vero e proprio. Attivandolo si accede al Main menu. Il Main menu offre tre opzioni fondamentali. Innanzitutto l'accesso agli otto editor che ci consentiranno la realizzazione del gioco. Poi la possibilità di mettere alla prova il lavoro fatto. Infine il salvataggio parziale o totale del lavoro sia come file provvisorio, passibile quindi di successivi interventi, sia come gioco vero e proprio, non più alterabile. Accediamo ora al primo degli editor, "Edit Sprites".

Edit Sprites

Tutti gli elementi in movimento nel gioco sono realizzati da questo programma mediante sprite. Con essi quindi daremo forma ai protagonisti, siano astronavi, oggetti, animali, alieni, proiettili o esplosioni. In un apposito sotto-menu sono raccolti gli otto comandi necessari al nostro lavoro. Con i primi quattro – Select Sprites, Edit Sprites, Select Colour e Edit Colour – possiamo generare sino a 127 sprite. Gli altri tre – Slide Sprites, Mirror Sprites, Copy Sprites ed Erase Sprites – ci consentono di eseguire movimenti o altre operazioni con gli sprite generati. Con il comando "Mirror Sprites", per esempio, ci è facile ruotare lo sprite generato, ottenendone così le diverse angolazioni. Queste, con l'opzione Edit Object che esamineremo più avanti, potranno servire per ottenere una buona animazione.

Edit Object

Perché gli sprite possano costituire un elemento effettivo del gioco dev'essere loro attribuita una funzione. Devono cioè diventare l'eroe del gioco, i suoi proiettili, un nemico, un proiettile del nemico, una sequenza animata di esplosioni... Nella terminologia di questo generatore devono diventare degli "Object", riconoscibili dal numero che viene loro assegnato. Ogni numero ha infatti un significato preciso: 00 indentificherà sempre il primo giocatore, 01 sarà sempre il proiettile del primo giocatore e così via.

Il sotto-menu Edit Objects presenta otto opzioni. Sette – Select Object, Edit Sprites and Place, Edit Color, Edit Anim Speed, Edit Anim Type, Test Object e Copy Object – servono alla costruzione degli object. I comandi per l'animazione sono molto raffinati. Con l'opzione "Edit Anim Type" possiamo dare luogo a un'animazione interessante per un massimo di 18 sprite. Con "Edit Anim Speed" possiamo in-

vece regolare la velocità di scorrimento dell'animazione sino a raggiungere l'effetto voluto.

Con l'ultima opzione del sottomenu, "Edit Enemy Bits", possiamo definire il comportamento nel gioco dell'object che raffigura il nemico. Possiamo così regolarne i parametri più importanti: la velocità di movimento, il punteggio che il giocatore ottiene colpendolo, il numero dei colpi necessari per l'eliminazione definitiva, il tipo di sparo (casuale o direzionale), la velocità dei proiettili e gli effetti sonori legati allo sparo e all'esplosione dei proiettili.

Edit Background

L'unità grafica di base utilizzata da questo generatore per costruire lo sfondo del gioco è un carattere di 4 per 8 pixel (Char). L'unità grafica superiore riunisce 5 per 5 caratteri (Block). Utilizzando così blocchi e caratteri siamo in grado infine di costruire la mappa complessiva del gioco (Map). Tutta questa strumentazione può sembrare eccessiva, ma i risultati che si ottengono sono entusiasmanti. Un'occhiata alle dimostrazioni e un'applicazione pratica eliminano ogni dubbio.

L'editor dispone di ben dieci opzioni. Con tre di esse – Select Char, Select Colour e Edit Colour – provvediamo a creare i caratteri necessari. Con altre tre – Select Block, Edit Block, Paint Block – otteniamo invece l'assemblaggio dei blocchi e quindi, con l'opzione Edit Map, inseriamo il tutto nella mappa del gioco. Le due opzioni rimaste – Copy Char e Copy Block – ci consentono di duplicare caratteri e blocchi secondo le nostre necessità.

Edit Sfx

L'ultimo tra gli elementi fondamentali del gioco è l'editor di effetti sonori. Mette a nostra disposizione un tool molto funzionale col quale siamo presto in grado di preparare gli effetti necessari al gioco. Due opzioni – Select Sfx e Edit Sfx – ci consentono per prima cosa di scegliere, tra i 24 disponibili, il suono su cui vogliamo intervenire, poi di modificare mediante il cursore forme d'onda, passo, attacco, velocità, decay e tempo. Il tool è completato da un'opzione di copy.

Dagli elementi alla struttura

Preparati gli elementi base, dobbiamo ora inserirli nella struttura del gioco. Il generatore prevede che prima siano definiti i ruoli dei giocatori, poi quelli dei nemici e infine che siano determinati i livelli di difficoltà. Esaminiamo le diverse operazioni.

Player Limitations

Questa opzione ci permette di definire i parametri che regolano il comportamento del giocatore. Possiamo fissare il numero dei giocatori abilitati, il numero delle vite a disposizione, la velocità di movimento, il numero dei proiettili disponibili, la loro velocità, la direzione del gioco, la gravità e le conseguenze delle collisioni. Con "Edit Player Area" possiamo delimitare l'area di movimento del giocatore; con "Edit Start Position" decidere invece la sua posizione di partenza. Le medesime operazioni devono essere ripetute per il secondo giocatore, se è previsto dal gioco.

Edit Attack Waves

Inseriamo ora nel gioco i "nemici", organizzandone il movimento e i percorsi. Si tratta di un'operazione delicata, richiede cura e tempo, e il consumo di memoria va costantemente tenuto sotto controllo.

Come per il giocatore, si tratta anche qui di decidere il luogo di partenza del nemico, tracciare il suo percorso e infine di memorizzare il tutto. Tramite l'apposita opzione, "Join Enemies", è possibile inoltre costruire "nemici" di dimensioni notevoli e di aspetto sorprendente.

Anche in questo caso si devono poi eseguire le operazioni necessarie a determinarne percorsi e comportamento.

Edit Levels

Il generatore permette di stabilire nel gioco sino a 22 livelli di difficoltà. Una prima opzione, "Edit Level Parameters", ci consente di scegliere per ciascun livello il relativo tipo – statico, a scorrimento automatico o controllato – la velocità di scorrimento, la durata della pausa tra un livello e un altro e quindi cosa deve accadere alla fine di ciascun livello. Possiamo far riapparire il medesimo schermo o far continuare il gioco con il livello successivo. Con l'opzione "Edit Level Map" possiamo infine fissare sulla mappa del gioco l'inizio e la fine di ciascun livello.

Un'ultima opzione, "Edit Front End", aggiunge un tocco di professionalità al gioco, consentendoci d'inserire un breve messaggio di presentazione. Opzioni successive ci consentono di variare i caratteri utilizzati e scegliere, tra 23 combinazioni di colori possibili, quella a noi più congeniale.

È giunto così il momento di provare il nostro gioco con l'opzione "Test Game" del Main Menu, e, dopo le opportune correzioni, salvarlo in via definitiva con l'opzione "Save Finished Game".

Osservazioni

Shoot'em up Construction Kit è decisamente il generatore di giochi che più si avvicina agli standard professionali. Ha suscitato larghi consensi, ottenendo addirittura rubriche esclusive su riviste straniere e italiane, che ne promuovono l'impiego da parte dei lettori.

Adventure Construction Set

Come abbiamo detto all'inizio dell'articolo, questo programma costituisce per gli utenti dell'Amiga un primo approccio ai generatori di giochi. Il suo autore, Stuart Smith, nell'ottimo manuale che accompagna il tool, paragona il suo lavoro a un immenso teatro che mette a disposizione degli utenti innumerevoli paesaggi e personaggi per la creazione

d'infinite storie. Non ci sembra inutile evidenziare il parallelismo esistente tra quanto si propone Smith e il ruolo che svolge il Dungeon Master nell'ormai notissimo *Dungeon & Dragons*, capostipite di tutti i giochi di ruolo.

Il Dungeon Master infatti, armato della sua creatività e delle indicazioni di massima fornite dalla struttura stessa del gioco, costruisce un mondo, vi ambienta una storia con eroi, mostri e tesori, e invita i suoi amici a entrarvi. Non diversamente si comporterà l'utente di questo programma. Anche lui sarà alle prese con un mondo da creare, completo di eroi, mostri e tesori. Alla fine avrà in mano un ottimo gioco di ruolo non dissimile dai più famosi esempi di questo genere, come *Phantasie III*, *Ultima IV* e simili.

Prima di esaminare il funzionamento del programma, è certamente utile descrivere la struttura dei giochi che è in grado di costruire.

Il mondo del gioco

La mappa del gioco si estende su un'area di 40 per 40 "square" (uno "square" ha dimensioni pari a quelle di una tradizionale icona del *Workbench*). Durante il gioco, solo un'area di 10 per 15 square sarà di volta in volta visibile sulla parte superiore dello schermo. Con lo spostarsi del giocatore, infatti, la mappa scorre dandoci una sensazione di movimento.

Disseminate sulla mappa ci sono poi le "porte" che danno sulle "regioni" o zone secondarie (insiemi di caverne, segrete, labirinti...) in cui si articola il gioco. Una "regione" può contenere sino a 16 differenti stanze o locazioni. Quando il giocatore imbocca una porta, si opera lo scorrimento e il disegno della nuova locazione sostituisce completamente quello precedente.

I giocatori

All'avventura possono partecipare sino a quattro giocatori per volta, individuati ciascuno da un'icona e da un nome. Ogni giocatore muove il proprio personaggio spostando il mouse o il joystick nella direzione desiderata. Il turno di ciascun giocatore è segnalato dalla "Movement Bar" posta a sinistra della mappa. A mano a mano che il giocatore muove il suo personaggio, l'altezza della "Movement Bar" diminuisce. Quando la barra è a zero, il tempo disponibile per il movimento è terminato. L'altezza della barra diminuisce gradualmente anche se il giocatore non muove il suo personaggio. Il giocatore infatti ha a sua disposizione un limitato tempo di gioco.

Si apre il sipario sulla scena di Adventure Construction Set: inizia l'avventura

Strettamente collegata con il movimento è l'acquisizione degli oggetti del gioco. Se il giocatore passa infatti sopra a un oggetto di cui si può impossessare, il sistema gli fornisce il nome dell'oggetto e lo informa dell'avvenuta acquisizione. Sempre il movimento determina i rapporti con le "creature" del gioco. Se infatti il giocatore muove il suo personaggio verso una "creatura" – controllata dal computer o da un altro giocatore – ed è in possesso di un'arma, ha luogo inevitabilmente un combattimento. Il computer fornisce quindi, attacco dopo attacco, la situazione dello scontro.

Le opzioni del giocatore

Paesaggi, personaggi, mostri e oggetti trovano la loro visualizzazione nella parte superiore dello schermo, unitamente alle barre che informano sulle condizioni del giocatore. Nella parte inferiore invece vi è un'area epressamente dedicata ai messaggi e ai comandi di gioco. Una prima riga contiene il nome del personaggio a cui tocca giocare (che può anche essere controllato dal sistema). A fianco compare il nome dell'area, regione o stanza, in cui si trova. Nelle restanti tre righe troviamo poi i menu accessibili al giocatore tramite joystick o mouse. Esaminiamo le diverse opzioni.

Move. Elimina il menu e quindi muove il personaggio.

Rest. Migliora la situazione di forza vitale del personaggio.

Use Object. Per impiegare un oggetto magico che si sta portando con sé. A volte per attivare l'oggetto sarà invece necessaria l'opzione seguente, Drop Object.

Drop Object. Per depositare un oggetto.

Use Power. Per lanciare un incantesimo. Maggiore è la forza vitale del personaggio, maggiore è la possibilità che l'incantesimo funzioni.

New Weapon. Per preparare all'uso un'arma che il personaggio sta trasportando.

New Armor. Per indossare o preparare una parte di armatura che il personaggio ha con sé.

Profile. Per esaminare i punteg

gi del personaggio, le sue varie caratteristiche e gli oggetti in suo possesso: armi o incantesimi.

Add Someone. Per aggiungere un nuovo personaggio all'avventura.

Retire. Per salvare il proprio personaggio e quindi usarlo più tardi in una nuova avventura.

Save Game. Per salvare la posizione raggiunta in un'avventura, se si vuole continuare il gioco in un altro momento.

Life Force Bar e Power Bar

La Life Force Bar, posta sulla destra della mappa, indica la resistenza del personaggio al combattimento. I colpi del nemico che raggiungono il personaggio gli sottraggono sia "power" che "life force". Così reciprocamente accade per i colpi che il personaggio mette a segno sul nemico. Quando la "Life Force Bar" è a zero, la battaglia è terminata e il personaggio è morto.

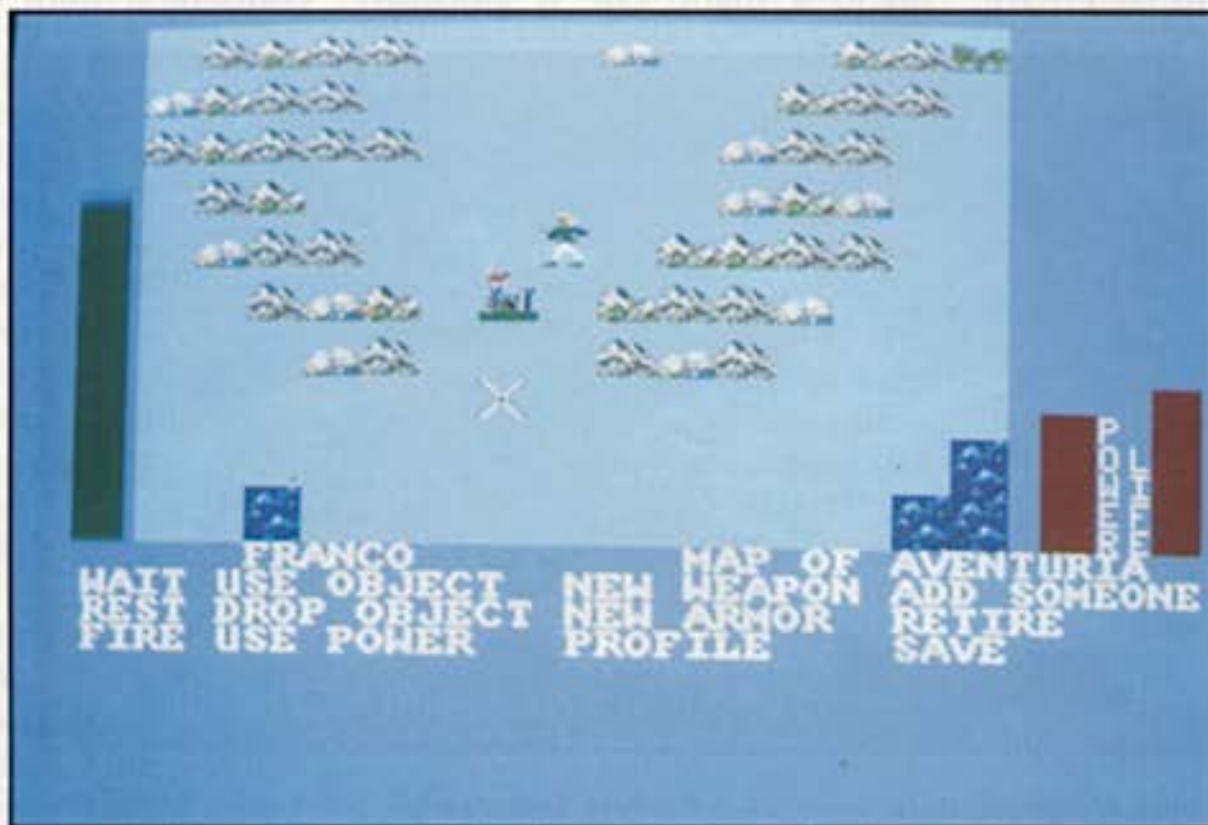
Durante la battaglia è possibile ripristinare la "life force" in due modi: usando l'opzione "rest" o trovando particolari oggetti che agiscono sulla forza vitale grazie alla magia.

La Power Bar, collocata anch'essa a destra della mappa, indica invece quanta energia magica il personaggio possiede. La "power" occorre sia per proteggersi nelle battaglie, sia per lanciare incantesimi.

Esaminati così i singoli elementi, vediamo il funzionamento del gioco.

Il generatore e i suoi menu

Il programma mette a disposizione dell'utente tre differenti set d'immagini e di strutture predefinite. Il primo è adatto alla costruzione di giochi di fantasy, il secondo a spy story o mystery, un terzo alla science fiction. È questa la prima scelta a cui ci troviamo di fronte.

Se selezioniamo pertanto dal menu generale l'opzione "Make an Adventure Disk" e dal relativo sotto-menu l'opzione relativa al set desiderato – Fantasy Builder Set, Spy/Mystery Builder Set, Science Fiction Builder Set – il programma provvederà a copiarla sul disco di lavoro o "adventure disk".

Ecco una fase dell'avventura dimostrativa creata con Adventure Construction Set

Selezionando poi l'opzione "Construct an Adventure", arriviamo al primo schermo di lavoro. Qui sono contenute le opzioni che ci permettono di lavorare sulle sovrastrutture della nostra avventura: titolo, nome dell'autore, tema musicale scelto ed eventuale introduzione. Vi sono inoltre tre opzioni che ci consentono di operare sul disco: salvare quanto abbiamo fatto, ritornare allo stesso lavoro, salvare in modo definitivo il gioco.

Un'apposita opzione consente infine di affidare al sistema l'autocostruzione dell'avventura. Questo comando costituisce una scorciatoia. L'opzione "Let ACS Finish Your Adventure" affida infatti al sistema la generazione di avventure basate su strutture casuali presenti nel programma. I risultati risentono della standardizzazione, ma, nella nostra esperienza, non sono mai ripetitivi.

Se tralasciamo questa possibilità e scegliamo invece l'opzione "Do More Detailed Work", giungiamo al secondo schermo di lavoro. Questo ci permette di lavorare sulla geografia del nostro gioco, sulla "World Map", sulle regioni e sulle stanze della nostra avventura. Possiamo cioè, servendoci di sotto-menu, creare e popolare la mappa, le regioni e le stanze con le diverse "creature". A partire da qui possiamo anche situare le "porte" che conducono da un luogo all'altro del gioco.

Anche il secondo schermo di lavoro presenta un'opzione "Do More Detailed Work", passiamo infine al terzo schermo. Questo ci permetterà di creare e modificare gli "oggetti", le "creature" e le immagini di cui abbiamo bisogno per la nostra mappa e le nostre stanze. Il menu contiene anche un'opzione che ci consente di cancellare una parte dell'avventura oppure di farcene una copia.

Il generatore in funzione

I menu che abbiamo visto finora ci consentono l'accesso agli strumenti per poter lavorare sulle cinque fondamentali parti del sistema ACS. La World Map contiene infatti i diversi tipi di terreno e le "porte" che si aprono verso (e da) le varie regioni della mappa. La Region Map esibisce invece le mappe di ciascuna regione, composte prima da un diagramma di tutte le stanze, poi

immagini di ciascuna stanza in dettaglio. La Thing List ci permette poi di attingere all'elenco di tutti gli oggetti. La Creature List offre i record di ciascuna creatura a cui si può accedere per popolare il nostro mondo. Il Graphics Set infine permette di selezionare le immagini necessarie per terreni, oggetti e creature.

Vediamo ora con un esempio pratico come ci si può servire di questa strumentazione.

In pratica

Cominciamo con la World Map. Dal secondo menu di lavoro selezioniamo l'opzione "Edit World Map" e quindi dal menu successivo l'opzione "Draw World Map". Subito compare un'ampia finestra entro cui lavorare. Al suo fianco un'icona con l'immagine del tipo di terreno che può essere impiegato per costruire la mappa. Più sotto il nome del terreno adottato in quel momento. Usiamo l'opzione "Select New Terrain" per informarci sui 16 tipi di terreno che abbiamo a disposizione. Gli ultimi tre, dal 14 al 16, rappresentano delle "porte". Per trasferire nella mappa il tipo di terreno che ci interessa non abbiamo che da compiere pochi movimenti col cursore e il nuovo terreno sostituisce quello precedente.

Ulteriori informazioni sono a disposizione dell'utente nella "World Map Reference Guide" dell'indispensabile manuale.

Arrivano le "Creature"

Nella World Map possiamo far comparire sino a otto creature. Selezioniamo quindi dal secondo schermo di lavoro l'opzione "Edit Map Creature" e poi "Add Another Map Creature". Comparirà un elenco del primo gruppo di creature disponibili nella Master Creature List. Spostiamo il cursore tra le diverse creature elencate e selezioniamo quella che ci interessa. Spostiamo ancora il cursore per selezionare un diverso gruppo di creature e sceglierne una adatta ai nostri scopi. Prima d'immetterla nel gioco, decidiamo dove la "creatura" dovrà apparire e con quale frequenza. Si possono anche controllare le caratteristiche della creatura scelta ed eventualmente modificarle. Basterà selezionare "Edit Traits". Comparirà uno schermo molto simile a quello relativo alle caratteristiche dei personaggi. Con il cursore possiamo modificare i valori numerici che esprimono queste qualità.

Regioni e stanze

Sappiamo che le avventure generate con questo programma possono avere sino a 15 regioni e che in ogni regione non ci possono essere più di 16 stanze. Dobbiamo ora decidere il numero delle regioni del nostro gioco e la quantità, le dimensioni e la posizione delle stanze. Per le regioni non abbiamo che da accedere all'opzione "Add region" dal secondo schermo di lavoro. Lì indichiamo il nome della regione, premiamo il pulsante e impieghiamo l'opzione "Add room" finché non raggiungiamo il numero di stanze desiderato.

Creature e oggetti

Per ciascuna regione che stiamo preparando, abbiamo a disposizione sino a 16 creature dalla Master Creature List e non più di 500 oggetti dalla Master Thing List. Selezioniamo l'opzione "Edit Region", e subito dopo la regione che contiene la stanza su cui vogliamo lavorare. Poi l'opzione "Select Room". Comparirà così un menu con quattro opzioni. Seguendo le indicazioni avremo tra l'altro la possibilità di decidere se le creature devono essere "residenti" nella stanza, cioè fisse, oppure "random", cioè a comparsa casuale.

Per aggiungere o togliere creature o oggetti, invece, impieghiamo l'opzione "Edit Content of Room". Per avere informazioni sul contenuto della stanza selezioniamo "Examine Room". Usiamo "Add 1 Thing" per accedere alla Master Thing List da cui selezionare l'oggetto che ci interessa. Usiamo la "Add Creature" per accedere alla Master Creature List e scegliere la creatura che desideriamo.

Se necessario, ampie informazioni su creature e oggetti sono a disposizione nella Creature Reference Guide e nella Thing Reference Guide del manuale.

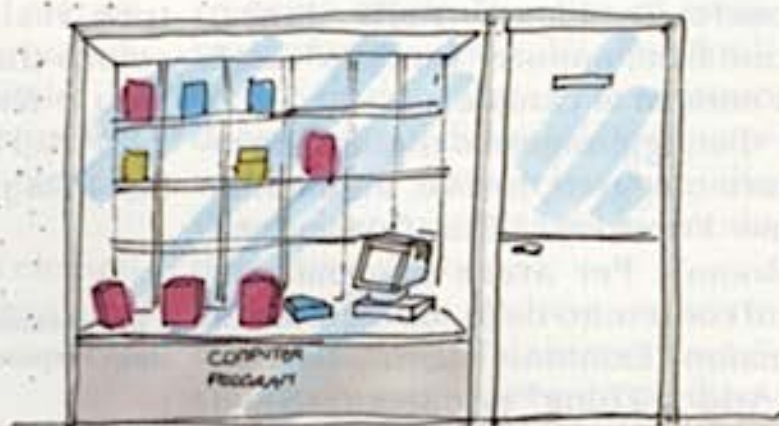
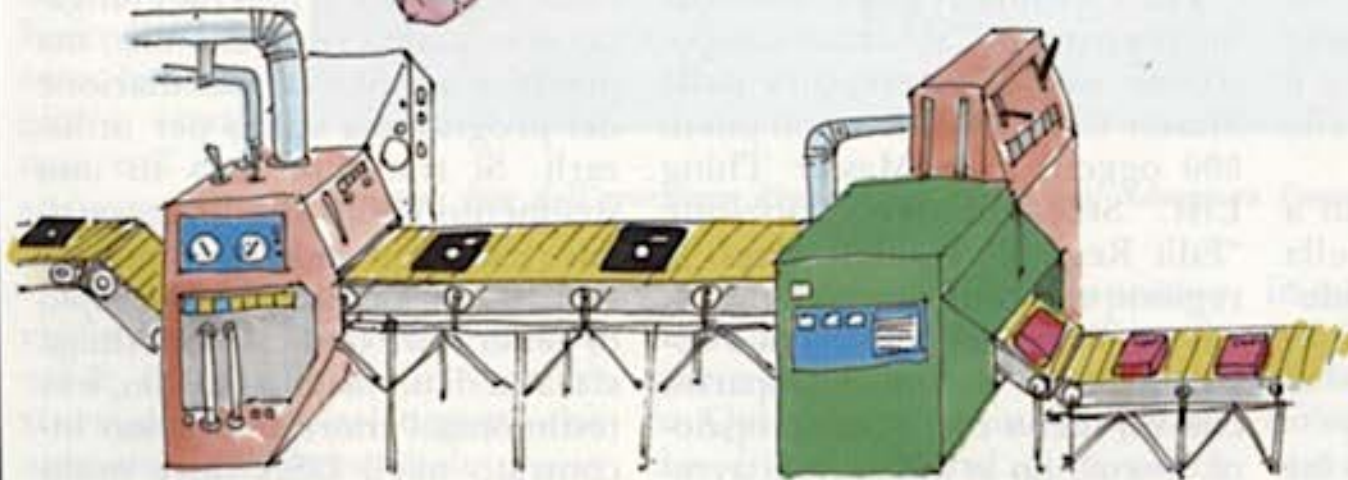
Due ultime possibilità completano gli strumenti a nostra disposizione. Possiamo aggiungere nuovi oggetti ai due Master. Ciascun Master contiene 128 possibilità. Per aggiungere un nuovo oggetto selezioniamo "Edit Thing" e quindi "Add Thing". Per le creature selezioniamo "Edit Master Creature List" e quindi "Create a Creature". Possiamo infine esaminare e modificare le immagini messe a nostra disposizione dal sistema, oppure crearne di nuove. A questo scopo è presente un'opzione apposita, "Edit graphics".

Osservazioni

Al contrario dei precedenti, questo programma non si può inserire nella categoria dei generatori veri e propri. Non perché i giochi costruiti con *Adventure Construction Set* non siano all'altezza degli altri, o non raggiungano una qualità professionale, ma perché è necessaria la mediazione del programma stesso per utilizzarli. Si tratta tuttavia di uno strumento molto ricco, capace di dare parecchie soddisfazioni a chi se ne serve. Come primo esempio di generatore dedicato all'Amiga si tratta di un ottimo esordio, e lo testimonia l'enorme successo incontrato negli USA dove molti appassionati hanno fondato club destinati allo scambio delle avventure realizzate. L'indirizzo per coloro che fossero interessati a un club è *Ken St. André, ACS Club, 3421 E. Yale, Phoenix, AZ 85008, USA*. Da provare. ■

I programmi descritti nell'articolo sono disponibili presso:

Lago snc
Via Napoleona, 16
22100 Como
(tel. 031/300174)

SOFTWARE HOUSE
CONTRATTO

COME FAR PUBBLICARE UN PROGRAMMA

I responsabili delle principali software house americane ci spiegano come viene effettuata la selezione dei programmi e dei programmatori

di Matthew Leeds

Avete scritto il miglior word processor, database o spreadsheet del mondo? Avete creato un'adventure o un arcade che tiene i giocatori incollati al video per una settimana intera? Il software che producete vi sembra innovativo o totalmente rivoluzionario? Probabilmente vorrete provare a vendere le vostre idee migliori a una software house.

Dal momento che anche chi scrive ha tentato di percorrere la stessa strada, peraltro senza successo, mi sembra interessante cercare d'individuare la via migliore per ottenere la pubblicazione di un programma. Proprio per questo motivo *Commodore Gazette* ha intervistato i responsabili dell'acquisto di software di quattro importanti aziende del settore: Patty Collins, Acquisitions Coordinator per la Broderbund; John Manley, Product Administrator alla Electronic Arts; Bob Lindsey, Director of Creative Development alla Epyx; e Richard Lehrberg, VP Product Acquisitions alla Activision.

Abbiamo registrato le loro opinioni, le abbiamo fuse con alcune osservazioni personali, e qui presentiamo i risultati di questo lavoro. Non possiamo assicurare che seguendo i consigli e le idee contenute in questo articolo otterrete sicuramente un contratto, tuttavia avrete almeno una buona base per cominciare con il piede giusto.

La prima cosa da fare, *prima* di contattare un editore, è valutare la qualità del vostro programma. È veramente il massimo che potete fare? È possibile migliorarlo? Provate a mostrarlo agli amici, fateglielo usare, giocateci, eseguite dei test. Confrontatelo con programmi dello stesso tipo già presenti sul mercato. Se non è almeno allo stesso livello, ritornate al tavolo da lavoro e tentate di migliorarlo.

Patty Collins: «Ci interessano prima di tutto prodotti che siano di livello superiore a quelli già esistenti, o che coprano aree di mercato ancora non sfruttate».

Richard Lehrberg: «L'errore che fanno spesso i programmatori è quello di emulare prodotti già presenti sul mercato. Gli editori cercano nuove idee».

Quando finalmente siete certi che il programma è quanto di meglio si può fare e che è al di sopra degli standard industriali, bisogna iniziare a considerare altri fattori. Ogni anno vengono pubblicati solo pochi programmi. Per la maggior parte vengono realizzati in diverse versioni, per poter girare su tutti i tipi di computer a larga diffusione. Gli editori, quindi, vogliono programmi che possano essere trasferiti da un computer all'altro e in linea di massima desiderano che siano scritti in Assembly o in C. Molto raramente vengono accettati programmi scritti in Basic o Basic compilato. Anche le utility che girano su un solo computer vengono accettate raramente, a meno che non si tratti di una macchina molto diffusa (come ad esempio il C-64).

Bob Lindsey: «Non ci piace acquistare prodotti dedicati a un solo sistema. Abbiamo bisogno di qualcosa che sia applicabile dappertutto. Non necessariamente i programmi in Basic vengono scartati, ma preferiamo l'Assembly perché gestisce meglio le nostre schermate grafiche».

Se invece avete un'ottima idea per un programma, ma non siete in grado di svilupparla personalmente, potete cercare un programmatore che se ne occupi al vostro posto. Per la maggior parte, gli editori con cui ho parlato preferiscono avere una versione funzionante del programma da esaminare. Certo, questo non

proprio un imperativo categorico, ma sicuramente le chance si moltiplicheranno seguendo questa indicazione. D'altra parte non è necessario che la versione presentata sia quella definitiva.

Patty Collins: «Cerchiamo prodotti che siano già in forma di programma. È molto più facile valutare un prodotto già funzionante, anche se non è completo. Se è possibile osservarlo direttamente e provarlo, il giudizio è più accurato e significativo».

John Manley: «La Electronic Arts è un po' diversa dagli altri proprio per il fatto che noi non siamo produttori, siamo solo editori. Parte del sommario che forniamo ai nostri programmatori insieme al kit spiega questo concetto. Non incoraggiamo a presentarci versioni dimostrative; vogliamo sperimentare il programma direttamente».

Richard Lehrberg: «Stiamo lavorando su un prodotto ideato da una donna che è un'artista, più che una programmatrice. È anche una giocatrice appassionata e ha avuto un'idea eccezionale per un gioco. L'abbiamo messa in contatto con un programmatore e insieme stanno sviluppando il prodotto».

Bob Lindsey: «È molto difficile acquistare soltanto un'idea. Anche per problemi amministrativi cerchiamo di evitare contratti di questo tipo: non rientrano nella nostra politica».

Bisogna inoltre tentare di mettere bene in luce le caratteristiche che rendono "speciale" il vostro programma. L'industria non ha certo bisogno di un ennesimo clone di *Pac-Man*. Se non riuscite a trovare qualcosa di esclusivo nella vostra creazione, non ci riuscirà nemmeno chi la deve acquistare. Il grande software è come la grande letteratura: il critico più severo dev'essere lo stesso autore.

John Manley: «Il computer porta qualcosa di nuovo nel gioco. Un programma di scacchi diventa più interessante sul computer in quanto si può giocare contro il computer stesso, si può tornare indietro di alcune mosse, oppure analizzare i movimenti sulla scacchiera. Ci sono casi, invece, in cui il programma è la semplice trascrizione per computer di un gioco già noto, senza alcun miglioramento. Noi cerchiamo qualcosa di innovativo, qualcosa a cui si possano agganciare altre idee».

Una volta superate queste fasi, se siete ancora sicuri di voler proporre il programma o l'idea, è tempo di decidere a quale editore sia meglio sottoporre il progetto. Ogni editore di software ha un proprio kit esplicativo per l'acquisizione dei diritti. Generalmente contiene una sommaria esposizione del contratto standard, un accordo di non-divulgazione che protegge il vostro programma da usi non autorizzati e un catalogo di prodotti. Studiate attentamente il catalogo e cercate di farvi un'idea del genere di programmi che ogni editore pubblica.

Patty Collins: «Devono assolutamente rendersi conto di quali prodotti ci interessano, e il modo più semplice è vedere quello che abbiamo pubblicato finora. Devono anche tenere conto della situazione di tutto il mercato».

John Manley: «Molti di coloro che ci sottopongono le loro realizzazioni sono nostri clienti. Chiamano al telefono e dicono: "Ho i vostri prodotti e mi piacciono moltissimo e ho un'idea per un buon gioco"».

Ora che avete i kit esplicativi, avete letto i cataloghi e avete deciso a quale editore affidarvi, è tempo di preparare un package per inviare il programma e farlo esaminare. È importante contrassegnare il disco col nome del programma, il tipo di macchina su cui gira, il vostro nome e indirizzo. Inserite istruzioni su come far girare il programma, un'eventuale documentazione e un contenitore già affrancato con il vostro indirizzo per la restituzione del package. Quasi tutti gli editori con cui ho parlato prendono in considerazione anche prodotti già in visione, perciò se desiderate inviare il package a più di un editore fatelo pure.

Patty Collins: «Preferiamo essere i primi a esaminare il prodotto, perché in questo modo abbiamo la "prima scelta", ma dipende dalla persona che ce lo sottopone. Alcuni ci inviano anche lettere di altri editori ai quali hanno sottoposto il proprio prodotto».

Richard Lehrberg: «Alcuni prodotti mi sono stati presentati con allegata una videocassetta dell'azione di gioco. Ciò può essere molto utile se ci sono livelli più alti che richiedono molto tempo per essere raggiunti».

Sono sicuro che tutti si stanno chiedendo cosa succede al proprio programma una volta che è stato spedito. Ogni editore riceve da 30 a 50 proposte circa ogni mese, e queste vengono esaminate una per una. Spesso i programmi vengono provati da più di una persona. Nella maggior parte dei casi viene redatto un giudizio scritto dai vari esaminatori.

Patty Collins: «Quando arrivano i programmi, faccio io una prima scrematura. In un secondo tempo abbiamo una riunione editoriale, di solito ce n'è una alla settimana, nella quale ognuno esamina i programmi che ho ritenuto interessanti. Se un programma ottiene un consenso generale, procediamo con le valutazioni. In seguito il programma viene visto da moltissime altre persone, e tutti danno il proprio giudizio».

John Manley: «Quando ci viene inviato un programma, lo guardo io per primo. Se ritengo che sia buono, lo passo a elementi dell'azienda che sono interessati a quel particolare tipo di software, e ne chiedo una valutazione. Per la valutazione, ci serviamo di un modulo prestampato, perché vogliamo giudizi scritti. Devono essere valutati sia gli aspetti positivi sia quelli negativi. Insomma, faccio in modo che ogni prodotto che ci viene sottoposto sia esaminato almeno da una persona».

Se il vostro programma non viene accettato, non abbandonate tutte le speranze. La risposta che vi verrà data al termine d[illegible]

cesso di valutazione potrebbe darvi utili indicazioni per migliorare il vostro lavoro, e poi è sempre possibile riproporlo.

Patty Collins: «A volte ci giungono lavori che sembrano avere grandi possibilità, ma che dovrebbero essere maggiormente curati sul piano della grafica e del design, e noi in quel momento non possiamo occuparcene, così li rimandiamo indietro con incoraggiamenti e suggerimenti».

John Manley: «Se decidiamo di non acquistare un programma, mi occupo io personalmente di prendere contatto con l'autore e gliene spiego i motivi. Inoltre faccio un elenco delle caratteristiche che ci sono piaciute e valuto il potenziale di sviluppo. Di solito il programma dovrebbe essere più curato. Unisco inoltre un elenco di critiche costruttive, segnalando ciò che a nostro parere potrebbe migliorare il prodotto oppure la direzione secondo cui dovrebbe essere modificato».

Richard Lehrberg: «Uno dei produttori o un membro della sezione sviluppo deve sostenere il prodotto. Se nessuno lo appoggia, è molto improbabile che venga pubblicato».

Se il vostro programma viene accettato, ha evidentemente superato la fase di selezione. D'ora in avanti produttori e editori ne seguiranno l'evoluzione lungo tutto il ciclo di sviluppo. Cosa accade poi?

Patty Collins: «Se decidiamo di accettare il programma, avviamo le negoziazioni con l'autore. Portiamo il programma all'interno dell'azienda. Se non è stato completato, ne portiamo a termine lo sviluppo. Ci sono alcuni marchi di riconoscimento che la Broderbund ha per i suoi prodotti, e di solito dobbiamo aggiungere alcune schermate grafiche o altre caratteristiche tipicamente nostre».

Bob Lindsey: «Immaginiamo che la Epyx voglia un programma. A questo punto proponiamo al programmatore di lavorare con il nostro staff. Vogliamo aumentare la sua esperienza per il prossimo prodotto che realizzerà. Si spera che la Epyx sarà la beneficiaria di questa esperienza ma non chiediamo a nessuno di prendere impegni esclusivi con noi. Diventa una vera e propria joint-venture. Abbiamo strumenti e utility che il programmatore può utilizzare per migliorare il proprio prodotto e per eliminare i problemi».

Ci sono altri modi per fare accettare i propri programmi, ma è questione soprattutto di essere nel posto giusto al momento giusto.

John Manley: «Ray Tobey incontrò Steve Wozniak (il fondatore della Apple, n.d.r.) al CES (la rassegna statunitense Consumer Electronics Show, n.d.r.), e gli mostrò il suo simulatore di volo. Wozniak scrisse una nota a Trip Hawkins, presidente della Electronic Arts, sul retro di un suo biglietto da visita che diceva "Questo è il miglior simulatore di volo che abbia mai visto" firmato *The Woz*. Disse a Ray di portarlo a Trip Hawkins e di dire che lo aveva mandato lui. È in questo modo che abbiamo ottenuto *Skyfox*».

I passaggi basilari sono semplici: create un programma che sia nuovo e diverso, tastate il polso al mercato, fatevi mandare i kit per l'acquisizione, seguite le istruzioni e incrociate le dita. Gli editori di software prendono volentieri in esame i vostri lavori, ma si aspettano che voi facciate ogni sforzo per presentarli in modo professionale.

Riguardo al perché gli editori siano così attenti al software proveniente da nuovi programmatori, chiedetelo a Richard Lehrberg: «Bisogna baciare molte rane per trovare un principe!». ■

GLI INDIRIZZI DA CONOSCERE

Superati tutti gli ostacoli, seguiti tutti i consigli, abbiamo finalmente realizzato il nostro programma e siamo arrivati al passo finale: la spedizione. E ora a chi dobbiamo rivolgerci?

A completamento dell'articolo ci è sembrato opportuno inserire un indirizzario di base (da conservare con cura!), che può fornire un primo e utile orientamento per iniziare a muoversi sul mercato. Si tratta di un elenco ovviamente incompleto, ma che comprende alcune tra le maggiori case produttrici di software di Stati Uniti, Inghilterra, Germania e ovviamente Italia. Per completare questo elenco è sufficiente acquistare le più diffuse riviste straniere, oppure cercare indirizzi sui manuali del vostro software preferito. Ma se si è davvero certi che il proprio programma è eccezionale, tanto vale partire subito dalle software house più prestigiose: un rifiuto potrà comunque essere istruttivo, e un successo avrebbe un valore molto più grande.

ITALIA:
Ital Video, Via A. Volta 2/A, 40055 Castenaso
Lago, Via Napoleona 16, 22100 Como
Leader, Via Mazzini 15, 25020 Casciago

GERMANIA:
Data Becker, Merowingerstr. 30, 4000 Düsseldorf
Markt & Technik, Hans Pinsel Strasse 2, 8013 Haar Bei München

INGHILTERRA:
Electronic Arts, Langley Business Centre, 1149 Station Road, Langley - Nr.Slough, Berkshire SL38YN
Firebird, Wellington House, Upper St.Martin's Lane, London WC2H9DL
Precision Software, 6 Park Terrace, Worcester Park, Surrey KT4 7JZ

STATI UNITI:
Activision, 2350 Bayshore PKwy., Mountain View, CA 94043
Aegis, 2210 Wilshire Blvd., Suite 277, Santa Monica, CA 90403
Broderbund, 17 Paul Dr., San Rafael, CA 94903-2101
Cinemaware Corp., 4165 Thousands Oaks Blvd., Westlake Village, CA 91362
Electronic Arts, 1820 Gateway Dr., San Mateo, CA 94404
Epyx, 600 Galveston Dr., Redwood City, CA 94063
IHT Software, 2269 Chestnut Street, Suite 162, San Francisco, CA 94123
Infocom, 125 Cambridge Park Dr., Cambridge, MA 02140
Microillusions, 17408 Chatsworth St.,Granada Hills, CA 91344
Microprose, 180 Lakefront Drive, Hunt Valley, MD 21030
Mindscape, 3444 Dundee Road, Northbrook, IL 60062
SSI, 1046 N. Rengstorff Avenue, Mountain View, CA 94043

LA PROGRAMMAZIONE IN AMBIENTE GEOS CON GEOPROGRAMMER

Ecco finalmente disponibile un pacchetto di programmazione che funziona in ambiente GEOS e crea applicazioni GEOS compatibili. Riunisce tre strumenti di lavoro fondamentali: geoAssembler, geoLinker e geoDebugger

di Luca Giachino

Creare semplici routine in linguaggio macchina non richiede particolari strumenti di lavoro; sono sufficienti un buon programma monitor e una buona conoscenza della macchina a cui la routine è destinata. In questo ambiente di lavoro "essenziale" il programmatore entra in un rapporto molto immediato con il computer, calandosi nella sua ridotta capacità di ragionamento e comunicando tramite bit, byte, valori numerici assoluti, indirizzi di memoria... tutto tradotto rigorosamente in notazione esadecimale. Ma in questo ambiente così "diretto" è sconsigliabile dedicarsi alla creazione di un'applicazione complessa, perché addentrandosi nella stesura è facile perdere di vista il risultato complessivo che si vuole ottenere. Il compilatore Assembly costituisce uno strumento di programmazione molto più confortevole, dal momento che consente l'adozione di un linguaggio simbolico, di espressioni algebriche, e di molte altre utili convenzioni.

In quest'articolo esaminiamo il pacchetto *geoProgrammer*, creato dalla Berkeley per la programmazione in ambiente *GEOS*.

geoProgrammer

Questo pacchetto di programmazione Assembly riunisce tre strumenti fondamentali per i programmatori: il compilatore Assembly *geoAssembler*, il linker *geoLinker*, e il debugger *geoDebugger*. Il compilatore produce file rilocabili che il linker provvede a riunire per creare un'applicazione eseguibile. Il debugger consente poi d'inoltrarsi nei codici dell'applicazione per snidare gli eventuali errori che sempre accompagnano le realizzazioni in linguaggio macchina.

Il pacchetto *geoProgrammer* opera in ambiente *GEOS* e consente la creazione di ogni tipo di applicazione *GEOS* compatibile... ma non solo. È infatti perfettamente idoneo per creare anche applicazioni non compatibili con *GEOS*, e non teme alcun confronto con altri pacchetti dello stesso genere. Uno dei suoi punti di forza, oltre alla completa compatibilità applicativa con *GEOS*, risiede infatti nelle sue origini.

geoProgrammer è una versione adattata di un sistema di sviluppo che gira in ambiente UNIX e che la Berkeley Softworks impiega per creare applicazioni *GEOS* compatibili. Lo stesso sistema operativo *GEOS* è stato creato in questo ambiente. *geoProgrammer* riprende tutte le caratteristiche operative del pacchetto UNIX, salvo alcune procedure di debug che richiedono un'apposita struttura hardware, detta in-circuit emulator, che serve per sostituire temporaneamente la CPU con un processore esterno.

Normalmente le grandi software house impiegano sistemi di cross development per realizzare le applicazioni più complesse per personal e home computer: sistemi molto veloci e con una capacità di calcolo considerevole. È sulla base di uno di questi è stato realizzato *geoProgrammer*.

Il manuale in inglese che accompagna il pacchetto è molto esauriente (circa 300 pagine), ed è organizzato in modo chiaro. Ognuno dei tre tool viene analizzato dettagliatamente, con particolare attenzione all'ottimizzazione del lavoro. La struttura stessa del manuale, ricco di esempi e appendici, ben si adatta a facilitare la comprensione di tutti gli argomenti illustrati.

Nelle appendici sono ripor-

listati di tutti i file sorgente contenuti nel disco, delle macro-istruzioni e della mappa di memoria, cioè tutte le informazioni necessarie per iniziare a programmare. Le appendici riportano anche un comodo glossario di 135 termini, e un completo indice analitico suddiviso per argomenti e ordinato alfabeticamente all'interno di ogni singolo argomento.

Il manuale non si propone d'illustrare il sistema operativo *GEOS*, ma solo di fornire gli strumenti per programmare in questo ambiente. Ogni volta che viene citata una peculiarità di *GEOS*, come per esempio la particolare gestione dei diversi tipi di file e la loro organizzazione su disco, oppure viene nominata una routine del Kernel, il manuale consiglia il lettore di consultare la *Guida ufficiale alla programmazione di GEOS*, che in Italia è pubblicata dalla IHT Gruppo Editoriale, la casa editrice di *Commodore Gazette*. Nella versione italiana, la guida è stata completamente aggiornata per essere compatibile con le specifiche introdotte dal compilatore *geoAssembler*. Per programmare in ambiente *GEOS* con *geoProgrammer* la *Guida ufficiale alla programmazione di GEOS* è uno strumento di fondamentale importanza.

I tre tool che compongono il pacchetto svolgono mansioni diverse, ma complementari. Analizziamoli singolarmente.

geoAssembler

Il compilatore *geoAssembler* legge un file sorgente prodotto con *geoWrite* e provvede a trasformarlo in un file rilocabile. La compilazione viene effettuata in due fasi per risolvere tutti i riferimenti in avanti e indietro. Il testo contenuto nel file sorgente può utilizzare qualunque fonte carattere, stile e impaginazione. All'interno del file risiedono il codice simbolico, i commenti e le direttive per il compilatore. Oltre a queste parti che non introducono fondamentalmente alcuna novità, i file sorgente possono anche contenere figure generate con *geoPaint*, *IconEditor*, o provenienti da *Graphics Grabber*. Provvede poi *geoAssembler* a inserire nel file rilocabile i dati delle mappe grafiche che incontra nel testo. Queste possono raffigurare icone, elementi grafici o disegni, che normalmente dovrebbero essere inseriti in un programma tramite lunghe sequenze di direttive ".byte". Non dimentichiamo inoltre che *GEOS* gestisce mappe grafiche compattate secondo tre possibili formati, e quindi i disegni inseriti nelle applicazioni devono rispettare almeno uno di questi formati. In genere il programmatore deve provvedere a compattare autonomamente un disegno e inserirlo nel file sorgente tramite la direttiva .byte. Con *geoAssembler* questo problema non si presenta, dal momento che ogni immagine inserita in un testo con *geoWrite* è già compattata.

Si tratta di una caratteristica poco comune, che si rivela di straordinaria utilità per programmare in un ambiente tipicamente grafico come *GEOS*.

Durante la compilazione, *geoAssembler* è in grado di gestire fino a 1000 simboli, label e assegnazioni all'interno di uno stesso file, consentendo d'impiegare un massimo di 20 caratteri per ogni simbolo alfanumerico (vengono però ritenuti significativi solo i primi 8 caratteri). Per quanto riguarda le espressioni che vengono risolte in fase di compilazione, *geoAssembler* dispone di un parser aritmetico-logico decisamente complesso, grazie al quale il

grammatore può gestire espressioni matematiche anche molto elaborate. Tutti i calcoli vengono svolti a 16 bit, garantendo così la doppia precisione numerica. Gli operatori disponibili sono 26, fra cui alcuni poco comuni nei compilatori, come l'operatore modulo (riporto di una divisione intera). Il parser è anche in grado di differenziare le operazioni AND, OR e XOR fra "logiche" e su "singoli bit".

La compilazione può essere subordinata a condizioni logiche tramite le direttive condizionali. In questo modo è possibile diminuire il tempo necessario alla compilazione, e aumentare la flessibilità dei file sorgente.

Inoltre *geoAssembler* consente la segmentazione della memoria in area di pagina zero, in area variabili RAM e in area codice. Questa segmentazione consente al compilatore di ottimizzare sia i modi d'indirizzamento sia l'allocazione in memoria del programma.

Oltre alle consuete direttive previste da quasi tutti i compilatori (fra cui la direttiva per l'inclusione di file che *geoAssembler* riconosce anche quando compare all'interno di un file incluso a sua volta), questo compilatore ne prevede altre espressamente realizzate per rendere più agile il compito del linker riducendo il numero di simboli che gli vengono passati. A queste si aggiungono alcune direttive necessarie per la compilazione del blocco File Header, il blocco che apre ogni file *GEOS* compatibile, e provvede a documentarlo con una serie d' informazioni utili al sistema operativo e alle applicazioni.

Per quanto riguarda le label, qualunque programmatore che impieghi abitualmente uno dei compilatori in commercio conosce bene gli sforzi che sono necessari per attribuire nomi unici e significativi alle etichette. Sforzi che sono spesso vanificati dall'inevitabile presenza di label prive di valore esplicativo, come quelle che caratterizzano i soliti loop. Per differenziarle da quelle significative e non obbligare il programmatore a sforzare troppo la fantasia, *geoAssembler* consente l'uso di semplici quanto funzionali label locali. Queste mantengono la loro unicità solo all'interno di una parte di codice delimitata da due label normali. Al di fuori di questa zona possono essere di nuovo impiegate senza creare conflitti.

Il desk accessory d'esempio creato con geoProgrammer

Il menu geos dell'applicazione VLIR proposta come esempio

Le macro istruzioni

Oltre a questo, *geoAssembler* è anche e soprattutto un macro-compilatore. Permette infatti di definire file di macro-istruzioni da includere nei file sorgente. "Macro-istruzione" è un termine che definisce un insieme d'istruzioni individuato da un nome simbolico. In linea di massima una macro-istruzione può anche essere priva di argomenti, ma di solito prevede il passaggio di alcuni parametri, che costituiscono appunto i suoi "argomenti". Quando un gruppo d'istruzioni si ripete, il programmatore può limitarsi a digitare il nome simbolico della macro-istruzione, indicando gli eventuali parametri. È compito del compilatore procedere all'espansione di tutte le macro citate nel file sorgente. In questo modo il testo sorgente diventa molto più semplice da digitare e guadagna in leggibilità.

Le macro-istruzioni possono diventare anche molto complesse, arrivando a contenere la quasi totalità delle direttive, comprese quelle condizionali. Possono accettare parametri multipli di ogni tipo, compresi modi d'indirizzamento e label, e nidificarsi fra loro, cioè contenere altre macro-istruzioni (cosa piuttosto rara in prodotti di questo tipo).

Il programmatore può definire macro-istruzioni secondo le più svariate esigenze. Volendo, è possibile definire un intero set d'istruzioni simulando l'Assembly di un altro processore, implementare complesse istruzioni che il 6510 non è in grado di svolgere, o generare nuovi modi d'indirizzamento. Per esempio, le istruzioni MoveB, MoveW, PushW, PullW, e molte altre, comuni in microprocessori evoluti, sono facilmente realizzabili tramite la macro-istruzioni.

La flessibilità di *geoAssembler* nella gestione delle macro-istruzioni è sicuramente un fiore all'occhiello che contribuisce enormemente ad accrescere la professionalità del prodotto.

Il file generato da *geoAssembler* si dice “rilocabile” in quanto gli indirizzi assoluti non sono ancora stati decisi. Si tratta quindi di un modulo rilocabile nel quale, eccettuati gli indirizzi assoluti, tutti i simboli non risolti sono assunti come esterni. Questo significa che il compilatore prepara autonomamente una tavola dei simboli non risolti da passare al linker, in modo che questi possa risolverli accedendo ad altri moduli rilocabili. Grazie a questa prerogativa, il programmatore non deve indicare i simboli che sono definiti esternamente, come accade con altri compilatori.

Se in fase di compilazione *geoAssembler* incontra un errore, genera un apposito file di testo che lo descrive dettagliatamente. La gestione degli errori compiuta da *geoAssembler* è molto raffinata. Sono previsti infatti 52 messaggi d'errore, ognuno ampiamente commentato nell'appendice.

In genere il programmatore compila diversi moduli separatamente, e provvede poi a conglobarli in un unico programma grazie al linker. Il linker è anch'esso uno strumento chiave quando si realizzano applicazioni complesse, e in un pacchetto di programmazione che abbia ambizioni professionali non può certamente mancare.

```
;*************************************************************************
;
;               SamVlirHdr
;
;       This file contains the header block definition for the GeoProgrammer
;       package sample VLIR application.
;
;Copyright (c) 1987 Berkeley Softworks. For the sole use of registered
;GeoProgrammer owners.
;*************************************************************************

if        Pass1                       ;Only need to include this file
noeqin                                ;during assembler's first pass.
include   geosSym                     ;get GEOS definitions
eqin
endif

;Here is our header. The SamVlir.lnk file will instruct the linker
;to attach it to our sample application.

        header                        ;start of header section

        word    0                     ;first two bytes are always zero
        byte    3                     ;width in bytes
        byte    21                    ;and height in scanlines of:

                              Vlir

        byte    $80 | USR             ;Commodore file type, with bit 7 set
        byte    APPLICATION           ;Geos file type
        byte    VLIR                  ;Geos file structure type
        word    ResStart              ;start address of program (where to load to)
        word    $3ff                  ;usually end address, but only needed for
                                      ;desk accessories
        word    ResStart              ;init address of program (where to JMP to)

        byte    "SampleVlir  V1.0",0,0,0,$00
                                      ;permanent filename: 12 characters,
                                      ;followed by 4 character version number,
                                      ;followed by 3 zeroes,
                                      ;followed by 40/80 column flag.

        byte    "Eric E. Del Sesto  ",0
                                      ;twenty character author name

        ;end of header section which is checked for accuracy
        block   160-117               ;skip 43 bytes...
        byte    "This is the GeoProgrammer sample "
        byte    "VLIR GEOS application.",0
        endh
```

Ecco come si presenta il testo sorgente di un File Header. Notate la presenza dell'icona

geoLinker

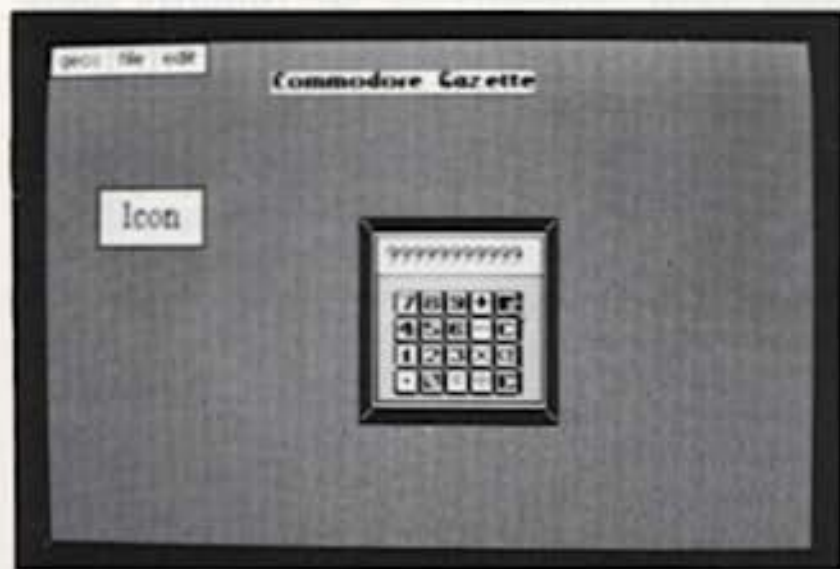
Nei pacchetti di compilazione evoluti il linker è uno strumento “strategico”. Dal momento che le sue funzioni sono ancora materia oscura per molti programmatori, descriviamole brevemente.

Quando si realizzano applicazioni molto lunghe, ricche di routine, alcune delle quali magari di utilità generale, è indispensabile suddividere l'intero programma in moduli. In questo modo è possibile trasformare provvisoriamente i moduli in piccole applicazioni separate per verificarne il funzionamento e localizzare più facilmente gli eventuali bug. Se poi il modulo è già stato impiegato altre volte, e si è dimostrato affidabile, diventa di estrema comodità conservare il file rilocabile ottenuto dall'ultima compilazione, per impiegarlo in seguito. Ovviamente l'applicazione deve provvedere agli opportuni riferimenti esterni a label e definizioni contenute nel modulo.

La funzione principale del linker è quindi quella di conglobare diversi moduli rilocabili in un unico programma, prendendosi cura di risolvere tutti i riferimenti incrociati esterni che in fase di compilazione l'assembler non ha potuto risolvere. Quando il linker lega fra loro i vari moduli, stabilisce anche come verranno allocati all'interno dell'applicazione, e di conseguenza nella memoria (salvo precise direttive contrarie).

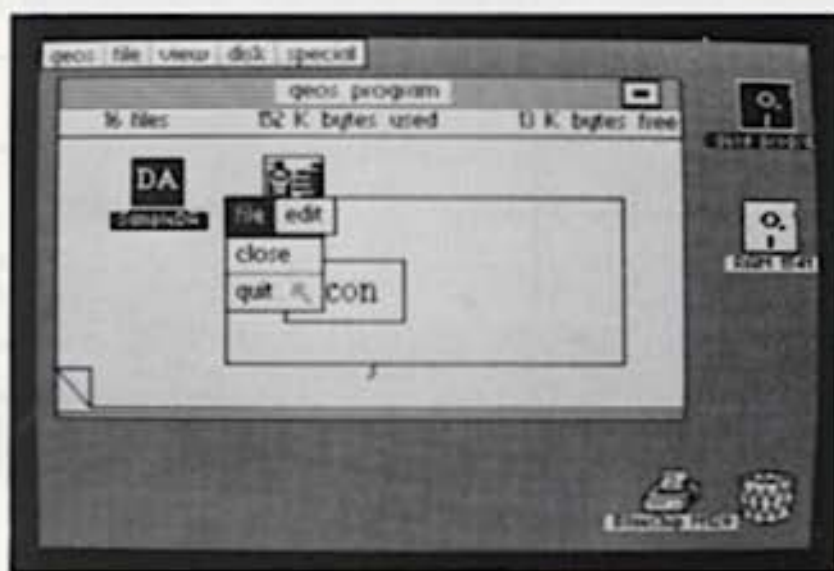
Queste sono in sostanza le funzioni che un normale linker deve assolvere. Con il tempo i programmatori possono accumulare una libreria di moduli già compilati e verificati alla quale accedere nello sviluppo di nuove applicazioni. Questi moduli sono rilocabili, e come tali non sono vincolati

a particolari aree di memoria.

Il linker incluso nel pacchetto, *geoLinker*, mantiene le funzioni standard appena descritte, e ne aggiunge altre di notevole rilievo. Per entrare in azione deve accedere a un file comandi opportunamente creato dal programmatore con *geoWrite*. Questo testo viene impiegato per indicare a *geoLinker* la struttura che dovrà avere l'applicazione, i moduli di cui è composta, se il blocco File Header è quello di default oppure è stato elaborato dal programmatore, il nome che deve avere il file eseguibile generato, e, nel caso di un file a struttura VLIR (struttura a Record Indicizzati di Lunghezza Variabile), prevede una precisa e chiara sintassi per indicare l'organizzazione logica dei moduli overlay dell'applicazione. All'interno di questo testo viene anche specificato se il file da creare dovrà essere *GEOS* compatibile, o solo un normale file Assembly. Nel secondo caso, infatti, *geoLinker* non include nel file il blocco File Header e non ne modifica il File Entry secondo le specifiche imposte da *GEOS*.

L'applicazione realizzata con geoProgrammer e Calculator

Accedendo alle informazioni contenute nel file comandi, *geoLinker* legge i moduli rilocabili ottenuti con *geoAssembler* e li congloba in un unico file eseguibile, prendendosi cura di risolvere tutti i riferimenti esterni incrociati (*geoLinker* è in grado di gestirne fine a 1700). I moduli rilocabili vengono trasformati in moduli non rilocabili risolvendo tutti gli indirizzi assoluti che *geoAssembler* non ha risolto. Effettivamente è anche possibile imporre a *geoAssembler* di valutare i riferimenti assoluti indicandogli, all'interno di ogni file sorgente, gli indirizzi per i vari segmenti del programma. Ma così facendo il file sorgente non è più un modulo completamente svincolato dalla posizione che assumerà in memoria, e non è più possibile disporre dello stesso modulo per altre applicazioni.

geoLinker è anche in grado di risolvere espressioni algebriche molto complesse contenenti riferimenti esterni (e quindi non valutabili da *geoAssembler*).

Quando *geoLinker* è in esecuzione, crea sul disco tre file, di cui uno opzionale. Viene creata l'applicazione eseguibile, un file contenente i riferimenti simbolici da impiegare con *geoDebugger*, e un file di testo, opzionale, contenente la lista completa di tutte le label e i simboli impiegati nella realizzazione dell'applicazione. Questo file è in pratica l'unico strumento di cui il programmatore dispone per controllare dove sono stati allocati in memoria i moduli e le routine del programma, dal momento che *geoLinker* non produce alcun file List. Sebbene questa non sia una grave limitazione, perché il pacchetto contiene *geoDebugger* per accedere ai codici dell'applicazione, è vero che in certi casi le applicazioni, specialmente se non sono *GEOS* compatibili, possono collocarsi in aree di memoria non raggiungibili con *geoDebugger*. In questi casi, un file List con il quale verificare l'esatta compilazione e allocazione dei codici potrebbe diventare una necessità. Oltre a quelli descritti, *geoLinker* provvede anche a creare un particolare file se riscontra qualche errore nella fase di link. I

Il desk accessory è stato mandato in esecuzione da deskTop

messaggi d'errore previsti, sono 25 e sono ampiamente descritti nell'appendice del manuale.

Terminata la fase di link, non rimane che sperare in una bassa presenza di bug e verificare il corretto funzionamento dell'applicazione con *geoDebugger*.

geoDebugger

Il debug è l'ancora di salvezza quando l'applicazione realizzata ha qualche difficoltà d'avvio. Si tratta di un programma che svolge funzioni molto complesse, ma che si dimostra flessibile e semplice da impiegare. Basta uno sguardo alle sue caratteristiche per intuire alle sue spalle la presenza di un sistema simile di più ampie dimensioni. La Berkeley Softworks ha infatti riprodotto in *geoDebugger* le funzioni disponibili in un sistema di debug che viene impiegato in ambiente UNIX.

Il debug è un processo quasi obbligato nella realizzazione di un programma in linguaggio macchina, perché le insidie nascoste in questo linguaggio sono molte e spesso difficilmente localizzabili.

geoDebugger non è un semplice programma monitor. È *anche*

efficace monitor, ma non solo. Raccoglie infatti altre funzioni che lo rendono uno strumento molto sofisticato.

Quando viene mandato in esecuzione, *geoDebugger* richiede all'utente d'indicare l'applicazione

Lo schermo iniziale di geoDebugger e le opzioni disponibili

che si desidera esaminare. Se viene indicato un file, *geoDebugger* scorre il disco alla ricerca del relativo file di simboli creato da *geoLinker*, e (se lo trova) lo carica insieme all'applicazione. Se invece non viene specificato alcun file, *geoDebugger* entra semplicemente nel modo operativo. È anche possibile mandarlo in esecuzione selezionando da *deskTop* un file di simboli generato da *geoLinker* (il nome termina con l'estensione .dbg).

L'ingresso nel modo operativo avviene in due modi diversi a seconda che sia presente o meno un'espansione di memoria. Nel primo caso *geoDebugger* carica da disco il Super-debugger e lo alloca interamente nell'espansione di memoria in un banco da 64K (l'impiego dell'espansione RAM da parte del Super-debugger non influisce sull'eventuale RAM disk installato). In questo modo l'utente può interagire con la versione più sofisticata di *geoDebugger*, che, sfruttando l'espansione, non interferisce minimamente con i codici del sistema e con lo spazio di memoria libero per le applicazioni. Se invece non è presente alcun tipo di espansione RAM, *geoDebugger* carica in memoria il Mini-debugger, una versione ridotta del Super-debugger che viene allocata in una parte dell'area disponibile per le applicazioni. Questa seconda versione presenta diverse limitazioni. La più appariscente è che non è trasparente al sistema, cioè deve convivere con il sistema e l'applicazione. Quindi per poter impiegare *geoDebugger* al massimo delle sue prestazioni, è necessario disporre di un'espansione di memoria.

Il debug di un'applicazione non è comunque l'unico impiego di *geoDebugger*. Questo programma è anche molto utile per esplorare il sistema operativo *GEOS*, verificare il funzionamento delle routine del Kernel e curiosare fra i codici senza che sia necessario realizzare ogni volta un'applicazione specifica.

Rivolgiamo ora la nostra attenzione al Super-debugger. Analizzeremo poi il Mini-debugger sottolineando i suoi limiti rispetto al Super-debugger.

Il Super-debugger

Quando si entra nel modo operativo, compare un normale schermo a caratteri con un cursore. Premendo F7 è possibile passare da questo schermo operativo a quello consueto di *GEOS* in alta risoluzione bit-map. In pratica il Super-debugger sfrutta un proprio schermo per comunicare con il programmatore, completamente indipendente da quello dell'applicazione. Il programmatore impartisce i comandi tramite un'interfaccia linea comando, e osserva quanto avviene attivando lo schermo di *GEOS*.

Quando *geoDebugger* assume il controllo del sistema, il Kernel di *GEOS* e l'eventuale applicazione sono come congelati. È quindi possibile visitare l'intero sistema, i vettori, lo stato delle variabili, alterare, intervenire, mandare in esecuzione routine... insomma, analizzare il flusso delle operazioni compiute dal Kernel e dall'applicazione.

Di solito quando si attiva *geoDebugger* si indica un'applicazione da sottoporre a verifica. Il debugger la carica in memoria (se è installato Super-debugger carica anche il file dei simboli), e attiva lo schermo operativo a caratteri cedendo il controllo al programmatore tramite l'interfaccia linea comando. A questo punto il primo passo consiste nel mandare in esecuzione l'applicazione e sperare...

Quando *geoDebugger* cede il controllo all'applicazione, viene attivato lo schermo di *GEOS*, e se non sono stati richiesti particolari interventi in fase di esecuzione dell'applicazione, la presenza del

Con geoDebugger è possibile disassemblare simbolicamente

debugger non è percettibile.

Il controllo viene restituito a *geoDebugger* in tre modi distinti. Se il programmatore preme RESTORE, il sistema viene congelato improvvisamente e il controllo ritorna a *geoDebugger*. Se invece si chiede all'applicazione, tramite la selezione di un menu o di un'icona, di tornare a *deskTop*, *geoDebugger* interrompe l'operazione

riprende il controllo.

Il terzo caso in cui *geoDebugger* riprende il controllo è il peggiore. Quando il processore incontra l'istruzione brk, il controllo torna a *geoDebugger*, e questo significa che c'è sicuramente un bug nei codici.

Le macro-istruzioni di sistema possono essere esplicitate

Quando si procede all'analisi dell'applicazione diventa molto utile una particolare prerogativa di Super-debugger, non prevista da Mini-debugger. Super-debugger è un debugger "simbolico", cioè è in grado di leggere un file di simboli e documentare i codici in memoria impiegando tutti i simboli preparati da *geoLinker*. Questa prerogativa è utilissima: si pensi alla comodità di disassemblare alcuni codici in memoria e leggerli come appaiono nel file sorgente, oppure di digitare un comando indicando le label e i simboli impiegati nella compilazione... È davvero molto comodo impartire il comando "dis ProgStart" per disassemblare una routine, invece che doversi ricordare l'indirizzo esadecimale corrispondente.

I comandi disponibili

geoDebugger mette a disposizione del programmatore un insieme di comandi molto ricco (64 comandi di base), che possiamo suddividere in diverse categorie.

Un gruppo di comandi è destinato all'esplorazione della memoria in modi e formati differenti. È possibile, per esempio, visualizzare il contenuto dei registri della CPU; a questo proposito è molto comoda l'espansione del PSW, che mostra con chiarezza lo stato di ogni flag. Oppure, dal momento che *geoDebugger* mantiene un PC (Program Counter) virtuale, è possibile richiedere di disassemblare i codici appena prima e appena dopo l'istruzione puntata dal PC. In questo gruppo di comandi riveste particolare importanza l'istruzione Print, grazie alla quale è possibile creare rapidamente qualsiasi comando di esplorazione dei codici. Grazie infatti al parser algebrico di cui dispone anche *geoDebugger*, e a una particolare sintassi, è possibile impartire comandi print molto elaborati, in grado di esplorare la memoria e visualizzarne i contenuti secondo qualsiasi formato, evidenziando simboli e stringhe ASCII.

Un altro gruppo di comandi è dedicato alla modifica dei codici contenuti in memoria. Per ragioni di spazio non è possibile descrivere esaurientemente le possibilità d'intervento offerte da questi comandi. Ci preme però sottolineare che il sistema interattivo impiegato da *geoDebugger* per alterare la memoria è estremamente pratico: selezionando il tipo di input/output con la semplice pressione di un tasto, consente d'inserire istruzioni, simboli, valori numerici nella base preferita, caratteri, espressioni, breakpoint, break-point condizionali...

Quando è in corso un input, il RETURN viene usato solo per segnalare la fine dell'inserimento dei dati, mentre per muoversi lungo le locazione di memoria si usano i tasti cursore. Per esempio, avendo inserito un'istruzione, per muoversi alla successiva locazione è sufficiente premere il tasto che muove il cursore verso il basso; se invece si desidera tornare indietro di qualche istruzione, basta risalire i codici con il tasto che muove il cursore verso l'alto.

Oltre a queste istruzioni, *geoDebugger* mette a disposizione un set di dieci comandi per l'esecuzione dei codici. Sono molto articolati e permettono di soddisfare qualsiasi esigenza, dal single-stepping al run-to-breakpoint. Consente inoltre di compiere diverse analisi sullo stack e sulla storia in esso contenuta. È per esempio possibile esaminare quali sono le ultime routine eseguite dal sistema.

Per il debug di un programma hanno molta importanza i breakpoint. *geoDebugger* mette a disposizione dell'utente diversi comandi molto articolati per servirsi dei breakpoint, alcuni dei quali consentono la definizione di breakpoint condizionali, cioè normali breakpoint che entrano in funzione in base al risultato di un'espressione. Questa espressione può assumere qualunque complessità, coinvolgendo registri, flag del PSW, periodicità, contatori, locazioni di memoria, simboli, e così via. *geoDebugger* dispone infatti di un modulo di calcolo algebrico molto evoluto, in grado di svolgere un grande numero di operazioni. In effetti i tipi di calcolo e gli operatori disponibili sono simili a quelli consentiti da *geoAssembler*.

I simboli per assemblare e disassemblare simbolicamente sono una delle caratteristiche più interessanti offerte da Super-debugger. Anche quando si sta assemblando una piccola routine di prova, è sempre utile poter lavorare con i simboli anziché gli indirizzi assoluti. In modo particolare, se l'applicazione è a struttura VLIR, i simboli sono raggruppati secondo i moduli a cui appartengono, ed è sempre possibile rendere attivo l'uno o l'altro gruppo.

geoDebugger, con il Super-debugger attivato, è anche un macro-debugger. In questo sistema, le macro sono comandi di base raccolti in un unico insieme caratterizzato da un nome simbolico

da alcuni argomenti. Molti dei comandi riconosciuti da *geoDebugger* sono macro-istruzioni che vengono chiamate macro-istruzioni di sistema.

Per creare un comando che svolga funzioni non ancora disponibili, è sufficiente seguire l'opportuna sintassi e costruirlo secondo le proprie necessità. In questo modo la quantità dei comandi disponibili è virtualmente infinita. Le macro-istruzioni di sistema sono un ottimo esempio; tramite un semplice comando è possibile richiedere al sistema la loro completa espansione, in modo da disporre di esempi particolarmente indicativi.

Le macro-istruzioni possono essere create tramite un file di testo proveniente da *geoWrite*, oppure direttamente con i comandi offerti da *geoDebugger*. Mentre nel primo caso sono sempre disponibili ogni volta che si attiva *geoDebugger*, nel secondo vengono perse al momento in cui il controllo ritorna a *deskTop*. Il programmatore può generare un file di macro-istruzioni da associare a un'applicazione particolare, ma anche un file slegato da ogni applicazione. È interessante notare che *geoDebugger*, quando carica questo file, cerca una macroistruzione di nome Autoexec, e se la trova la manda subito in esecuzione. Questa può contenere alcuni Prompt per l'utente, personalizzare il debugger tramite il comando Opt, e svolgere tutte le altre operazioni necessarie perché il programmatore, quando riceve il controllo, possa subito iniziare a lavorare nell'ambiente che preferisce, senza dover ripetere ogni volta la stessa noiosa procedura.

Il manuale è molto dettagliato su questo argomento e offre numerosi esempi applicativi.

Un altro set di comandi permette di compiere le normali operazioni di spostamento, riempimento, confronto e copia di aree di memoria.

Dal momento che il Super-debugger permette di assemblare e disassemblare simbolicamente, diventa comodo poter disporre di variabili interne temporanee per ricordare valori numerici e dati senza occupare locazioni di memoria. Il Super-debugger offre dieci variabili utente che possono essere impiegate in qualsiasi espressione algebrica. Queste, insieme alle variabili interne che individuano lo stato dei registri e dei flag del PSW, costituiscono un insieme di variabili che si rivela utilissimo nella creazione delle strutture condizionali e delle espressioni algebriche.

L'ultimo gruppo di comandi riguarda le interazioni con i disk drive. Dopo aver selezionato uno dei disk drive disponibili, è possibile leggere la directory del disco in esso contenuto. Questo comando è una macro-istruzione di sistema, e può quindi essere elaborato opportunamente per cambiare il formato e le informazioni visualizzate sullo schermo. Dal momento che *GEOS* gestisce molte informazioni aggiuntive per ogni file, è possibile strutturare il comando che visualizza la directory in modo che vengano mostrate le informazioni che interessano, nei formati più idonei.

Vi sono anche altri comandi dedicati alla gestione dei blocchi del disco. Tutti accettano come argomenti l'indirizzo traccia/settore, ma se questo non viene indicato, è necessario ricorrere ai due registri di *GEOS* di cui si servono abitualmente le routine d'accesso al disco per identificare gli indirizzi traccia/settore. È possibile prelevare un blocco dal disco, riscriverlo, accedere al successivo lungo un concatenamento, e visualizzare tutti gli indirizzi traccia/settore dei blocchi appartenenti a un concatenamento. Combinando opportunamente questi comandi all'interno di macro-istruzioni, è possibile creare nuove istruzioni in grado di salvare aree di memoria sul disco, di caricare file dal disco, e di compiere svariati altri compiti che l'insieme di comandi standard non permette.

Questa panoramica sui comandi disponibili con il Super-debugger non è certamente sufficiente. Il manuale è comunque molto completo, e per ogni comando sono descritti tutti i possibili formati, la sintassi e qualche esempio d'impiego. Già con i comandi standard, il programmatore dispone di funzioni superiori a quelle fornite da molti altri prodotti. Se a questo insieme vengono aggiunte tutte le macro-istruzioni che ogni programmatore può realizzare per soddisfare le proprie necessità, le possibilità d'impiego e la flessibilità del Super-debugger diventano davvero impressionanti.

Prima di passare a descrivere brevemente il Mini-debugger, citiamo un comando che riveste particolare importanza in un sistema operativo che risiede interamente su RAM come *GEOS*.

Durante il debug di un programma, o durante l'esplorazione delle routine del sistema, può accadere che un comando errato, o una routine mal funzionante, alterino il Kernel di *GEOS*. In questo caso, se il Kernel riprendesse il controllo potrebbe bloccarsi. Per ovviare a questo inconveniente, il comando rboot provvede a ricaricare l'intero Kernel dall'espansione RAM, ovviando così a qualunque eventuale alterazione del sistema causata da *geoDebugger*.

Il Mini-debugger

Il secondo programma che *geoDebugger* installa in memoria se non è presente nessuna espansione, ha molti limiti rispetto al Super-debugger, ed è più vicino a un programma Monitor che a un debugger.

Il limite più grande è che risiede in memoria, occupando una parte dell'area diponibile alle applicazioni. Come conseguenza, i comandi disponibili sono inevitabilmente molto ridotti.

Oltre a questo, il Mini-debugger non è in grado di assemblare e disassemblare simbolicamente, e non impiega le macro-istruzioni. È inoltre sprovvisto del sof

cato parser algebrico di cui dispone il Super-debugger, e consente d'indicare soltanto numeri esadecimali, e non espressioni.

Per quanto riguarda il modo operativo, anche il Mini-debugger adotta l'interfaccia linea comando per comunicare con il programmatore su uno schermo indipendente da quello di *GEOS* e impiega gli stessi metodi interattivi per l'alterazione della memoria: per svolgere veloci analisi o effettuare il debug di piccole applicazioni è uno strumento comunque superiore ai normali programmi monitor.

Particolari tecnici

Mettiamo ora in evidenza alcuni particolari tecnici del pacchetto che interesseranno sicuramente tutti coloro che impiegano abitualmente un compilatore Assembly per creare i programmi.

Nel pacchetto sono inclusi, oltre ai tre tool di cui abbiamo già parlato, diversi file utili al programmatore per familiarizzarsi con il pacchetto. I primi costituiscono tre esempi di file sorgente che il programmatore può compilare e trasformare in file eseguibili. Si tratta di un file a struttura SEQUENTIAL, uno a struttura VLIR, e un desk accessory. Con questi tre esempi il programmatore ha modo di verificare come si realizzano le applicazioni nelle tre strutture fondamentali. Purtroppo la versione di *geoProgrammer* in nostro possesso presenta un bug in *geoLinker* che non ci ha permesso di creare il file esemplificativo a struttura VLIR, mentre per gli altri due non abbiamo incontrato problemi. Abbiamo comunque simulato il lavoro del linker creando tre moduli eseguibili e conglobandoli nello stesso file VLIR, verificando così l'esattezza dei codici anche in questo caso.

I tre esempi mettono in evidenza alcuni interessanti aspetti di *GEOS*, come la gestione dei desk accessory e quella in overlay dei moduli nella struttura VLIR, che possono essere utili nella stesura delle proprie applicazioni.

Ma torniamo a *geoProgrammer*. Sul disco sono presenti anche i file di testo che riportano gli assegnamenti di tutte le costanti, le variabili e le routine descritte nella *Guida ufficiale alla programmazione di GEOS*.

Vediamo ora alcuni aspetti di *geoAssembler* che meritano particolare attenzione. Questo compilatore mantiene aggiornato un contatore per ognuno dei tre segmenti di cui è normalmente composto un file sorgente. Mantiene inoltre aggiornate tre variabili durante l'intera compilazione. Due riportano le dimensioni della figura che *geoAssembler* ha appena compilato, in modo che il programmatore le possa impiegare all'interno dei codici. Dal momento che alla figura successiva *geoAssembler* aggiorna nuovamente queste due variabili, è bene che all'interno del file sorgente il loro contenuto sia sempre assegnato ad altri simboli.

La terza variabile è molto utile per ottimizzare la compilazione. Si tratta di un flag che serve a distinguere la prima fase della compilazione dalla seconda, evitando così d'includere anche nella seconda fase della compilazione i file di sistema .include, con relativa riduzione del tempo di compilazione.

Sempre per ottimizzare il lavoro, *geoAssembler* consente d'individuare gli assegnamenti che vengono passati anche a *geoDebugger*. In questo modo si evita di appesantire inutilmente il carico di simboli che vengono impiegati durante il debug.

Il compilatore *geoAssembler* è uno dei più veloci compilatori Assembly mai realizzati. Abbiamo compilato 200 linee di codice con un normale Assembler in un minuto e 24 secondi. Lo stesso file sorgente è stato compilato da *geoAssembler* in 23 secondi, mantenendo un file di input e uno di output su disco.

Il tempo necessario alla compilazione di un file su disco è quindi già buono, ma viene ancora drasticamente ridotto al 30% se si compila il file sul RAM disk. La stessa cosa si si verifica con *geoLinker*.

Conclusioni

Per programmare in Assembly, questo pacchetto rappresenta sicuramente il miglior prodotto che sia mai stato presentato sul mercato per il C-64. Offre una grande flessibilità d'impiego e una serie di caratteristiche davvero eccezionali. La gestione delle macro-istruzioni, dei moduli in overlay, il debug simbolico, l'impiego delle espansioni RAM e le altre caratteristiche che abbiamo descritto fanno di questo pacchetto uno strumento di programmazione completo. Utilizzandolo insieme alla *Guida ufficiale alla programmazione di GEOS*, l'utente può creare applicazioni *GEOS* compatibili di qualsivoglia complessità.

Impiegando *geoProgrammer* abbiamo notato solo due fondamentali problemi. Il primo riguarda l'incapacità di *geoDebugger* di stampare il contenuto della memoria; questa ci sembra una grave limitazione. Il secondo, invece, è meno grave ma abbastanza pericoloso. Se *geoAssembler* e *geoLinker* creano un file per il quale viene utilizzato un nome già esistente, nella fase di salvataggio il file precedente viene sovrascritto senza avvisare l'utente, col rischio di perdere precedenti versioni che non si intendono cancellare.

Il prezzo di questo pacchetto, considerando che si tratta di un prodotto professionale che riunisce in sé tre distinte applicazioni, non ci sembra eccessivo.

I ferri del mestiere ora esistono, l'unico limite è la fantasia. ■

geoProgrammer Lire 100.000
DeskPack Plus Lire 60.000
sono disponibili presso:

Lago snc
Via Napoleona, 16
22100 Como
(tel. 031/300174)

DESKPACK PLUS

Questo pacchetto riunisce in un unico disco diverse applicazioni. La novità più importante che introduce è che tutte queste applicazioni sono compatibili sia con *GEOS*, sia con il modo 80 colonne di *GEOS 128*. Possono quindi essere caricate da *GEOS 128* senza dover passare al modo 40 colonne, per la gioia di tutti i possessori del C-128. La trasformazione effettuata per realizzare questa compatibilità è ampiamente documentata nella *Guida ufficiale alla programmazione di GEOS* edita dalla IHT Gruppo Editoriale, la casa editrice di *Commodore Gazette*.

Esaminiamo sinteticamente le applicazioni contenute nel pacchetto, mettendo in risalto le novità che introducono rispetto alle precedenti versioni.

Graphics Grabber consente di trasformare schermate prodotte dai più noti programmi grafici in file *GEOS* compatibili, in modo che possano essere impiegate con *geoWrite*, *geoPaint*, *geoPublish*...

Icon Editor converte file non-*GEOS* compatibili nel formato impiegato da *GEOS* (cioè, aggiunge il blocco File Header, che comprende anche l'icona). Permette inoltre di disegnare le icone associate ai file, in modo da personalizzarle secondo il proprio gusto e le proprie esigenze. Rispetto alle precedenti versioni, sono state incluse le opzioni *cut* e *paste* grazie alle quali è ora possibile generare le icone con i file prodotti da *geoPaint*. Ma non solo. Tramite queste opzioni è anche possibile generare un'icona e salvarla su disco come file prodotto da *geoPaint*, in modo da poterla poi richiamare all'interno di un testo redatto con *geoWrite*. Questa possibilità è molto utile per inserire le icone all'interno dei file sorgente creati per il pacchetto *geoProgrammer*.

Lo schermo che appare attivando l'applicazione Icon Editor

Calendar è un comodo accessorio da scrivania con il quale possiamo richiamare un'aggiornata agenda elettronica durante qualsiasi fase di lavoro. È stato ampliato con l'aggiunta di un'opzione che consente di prelevare testi prodotti da *geoWrite* e inserirli come promemoria all'interno dell'agenda.

Blackjack è semplicemente una versione per computer dell'omonimo gioco. Per i momenti di relax.

geoDex è un archivio di record che consente di memorizzare diverse informazioni per ogni nominativo. Ogni record può essere individuato molto rapidamente, anche svolgendo ricerche parziali. Inoltre l'applicazione *geoDex* è stata ampliata per consentire la copia del record corrente all'interno di un file Text Scrap, permettendone così l'impiego da parte di altre applicazioni, come ad esempio *geoWrite*. Permette anche di stampare liste di nominativi secondo diversi formati, compreso quello per le etichette.

geoMerge è un'applicazione che consente di stampare un documento base, creato con *geoWrite*, cambiandone particolari attributi ogni volta, in modo da automatizzare la stampa di una serie di lettere personalizzate. Per ogni copia del documento, *geoMerge* interroga l'utente sui testi che devono essere sostituiti ai simboli (attributi) presenti nel file-base, in modo che il destinatario del documento, ad esempio, sia diverso per ogni stampa. La procedura d'inserimento dei dati può essere anche automatizzata: *geoMerge* offre infatti la possibilità di leggere un file di record generato da *geoDex* per sostituire automaticamente a ogni simbolo la relativa stringa di testo.

Oltre a tutte queste applicazioni, il pacchetto contiene anche la versione 1.3 di *deskTop* e gli ultimi driver di stampa disponibili.

Analizziamo ora le applicazioni più importanti entrando maggiormente nei dettagli.

Graphics Grabber

Quest'applicazione consente di trasformare i file di dati grafici prodotti dalle applicazioni *PrintMaster*, *Print Shop* e *Newsroom*, in file *geoPaint* compatibili. Per esempio, *Newsroom* dispone di moltissimi disegni già realizzati che sono molto utili per decorare una pagina, ed è molto utile poterli impiegare anche in ambiente *GEOS*.

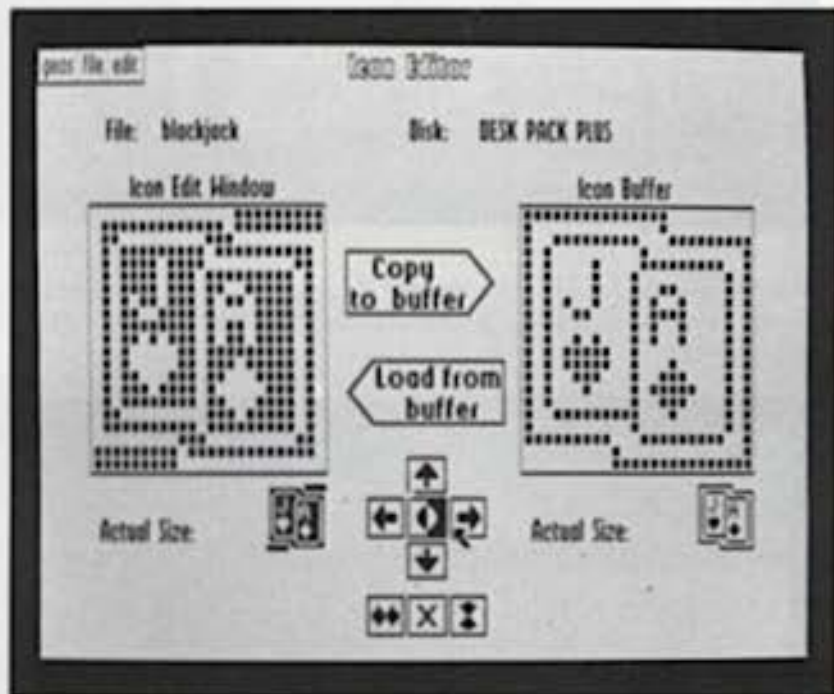
Graphics Grabber interagisce con l'utente tramite una barra menu e una serie d'icone. Alcune icone, come vedremo, non sono altro che metodi più rapidi per selezionare opzioni disponibili anche nei menu.

Quando l'utente, aprendo il menu file, ha indicato l'applicazione d'origine del file che desidera convertire, *Graphics Grabber* procede a visualizzare tutti i file di dati grafici che sono presenti sul disco. Una volta che è stato selezionato anche il file, il suo contenuto viene caricato e visualizzato sullo schermo.

A questo punto l'utente ha la facoltà di salvarlo all'interno di un file scrap o di un Photo Album. La prima scelta è temporanea, in quanto *GEOS* gestisce solo un file scrap alla volta, e se si trasferisce un altro disegno, questo andrà a memorizzarsi nello stesso file scrap. Invece, l'impiego del Photo Album permette di archiviare il disegno in modo permanente, rendendolo disponibile a qualsiasi altra applicazione.

Graphics Grabber permette di agire anche molto rapidamente, in modo da trasformare un grande numero d'immagini in poco tempo. Questa rapidità diventa possibile impiegando le icone. Per esempio, sullo schermo compare un'icona che rappresenta il tipico box di dialogo per la scelta dei file. Selezionandola ricompare la lista dei file presenti sul disco, in modo che l'utente possa rapidamente sceglierne un altro. Esiste inoltre un'icona che permette di scorrere avanti e indietro lungo i disegni, o di saltare al primo o all'ultimo della serie. Quest'icona assume due comportamenti diversi a seconda che il file di dati grafici provenga da *Newsroom* o dagli altri due programmi grafici. Infatti, quando si seleziona un file prodotto con *Newsroom*, l'icona scorre le varie immagini all'interno del file, mentre con gli altri due programmi scorre i file presenti sul disco.

Icon Editor

Gli impieghi di quest'applicazione sono diversi. Il più

evidente è la modifica, o la creazione, delle icone associate a qualunque file *GEOS* compatibile. Questo consente di personalizzare le icone secondo il proprio gusto, o in riferimento al particolare contenuto del file.

Quando però si richiede di creare l'icona per un file non-*GEOS* compatibile, *Icon Editor* procede a trasformare il file nel formato *GEOS*. Questa trasformazione comporta l'aggiunta di un blocco, denominato File Header, nel quale vengono memorizzate tutte le informazioni che *GEOS* aggiunge normalmente ai suoi file. All'interno del blocco File Header, per esempio, trovano posto l'icona del file e uno spazio libero impiegato da *deskTop* per inserire un commento.

La trasformazione si estende anche al File Entry del file presente nella directory, al quale vengono aggiunte alcune informazioni nei byte che non vengono impiegati dal Kernel di *GEOS*. Questi byte vengono impiegati dal 1541 per gestire i file random e i file relativi, due strutture che non sono riconosciute in ambiente *GEOS*.

Il terzo impiego di *Icon Editor* riguarda la creazione d'icone prelevando immagini da file scrap (di solito create con *geoPaint*), e la creazione di file scrap contenenti icone. Questa seconda opportunità si rivela molto utile quando si opera con *geoProgrammer* per inserire un'icona direttamente all'interno di un file sorgente creato con *geoWrite*.

Interagire con quest'applicazione è molto semplice. L'utente dispone di due buffer, uno principale e uno di sicurezza. Il secondo buffer è molto comodo per conservare una copia dell'icona originale durante una fase di modifica, oppure quando si desidera assegnare la stessa icona a più file. In quest'ultimo caso, quando si sceglie un nuovo file il contenuto del buffer di sicurezza viene copiato nel buffer principale, così che questo possa assegnare la stessa icona all'ennesimo file. Per modificare o creare le icone, *Icon Editor* le visualizza ingigantite all'interno di due finestre, una per buffer. Le operazioni che si possono compiere, oltre alla modifica dei singoli pixel, sono la rotazione verticale e orizzontale, l'inversione di tutti i pixel, e la traslazione orizzontale o verticale.

Sopra: lo schedario visualizzato da geoDex e le varie opzioni. Sotto: un esempio di testo da impiegare con geoMerge

geoDex

L'applicazione *geoDex* è un desk accessory studiato per la gestione di record contenenti nominativi, indirizzi, e altre informazioni che si devono poter richiamare istantaneamente, secondo diverse chiavi di ricerca, stampate e preparate per essere impiegate con *geoMerge*.

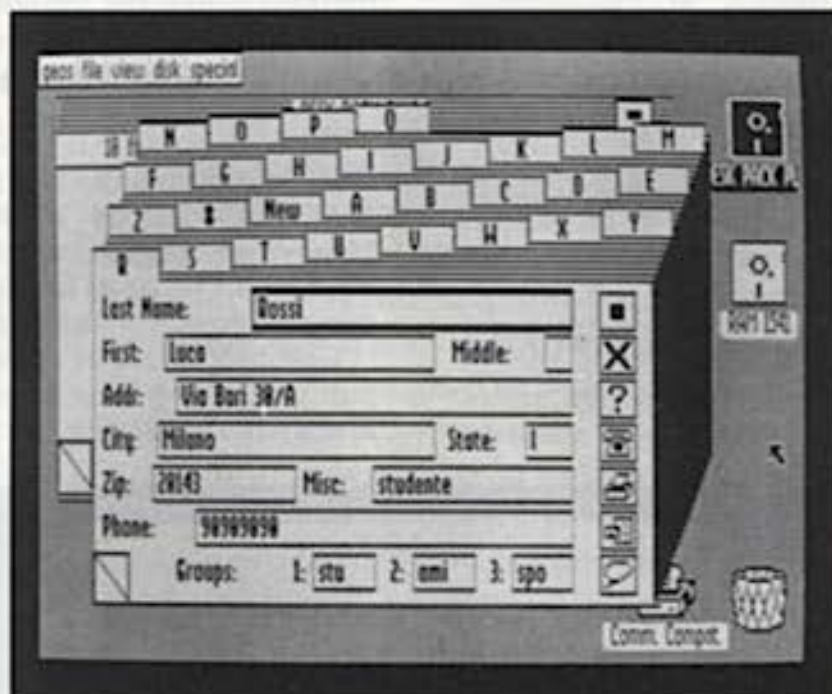
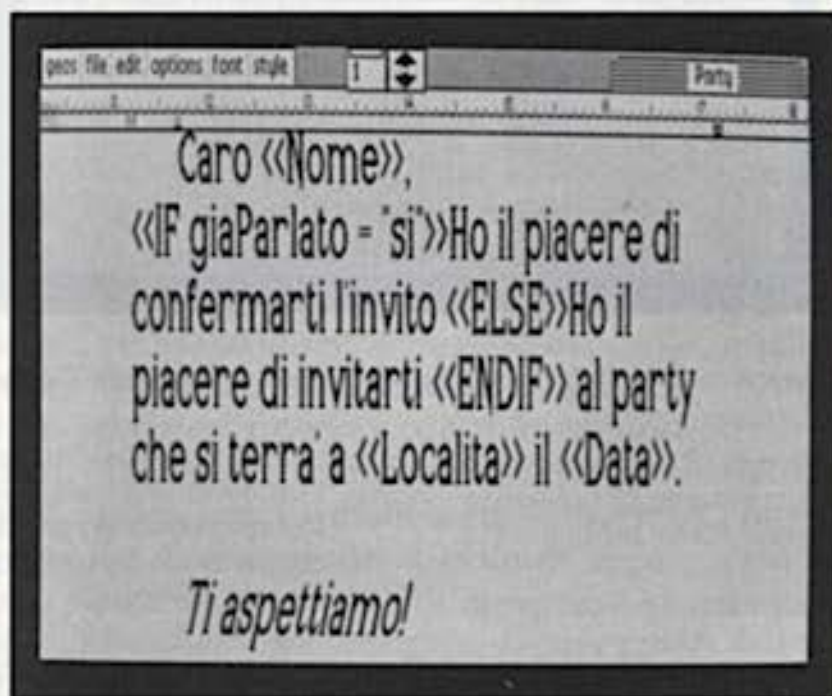
All'utente si presenta una specie di schedario nel quale ogni scheda è formata da diversi campi fissi. I campi sono: cognome, nome, iniziale del secondo nome, indirizzo, città, Stato, codice postale, numero telefonico, un'eventuale nota e infine tre campi per il gruppo di appartenenza. Questa serie d'informazioni viene associata a ogni nominativo.

L'utente ha quindi la possibilità di registrare un nominativo in ogni record, e stabilirne il gruppo d'appartenenza. Questo si rivela particolarmente utile per differenziare, ad esempio, i nominativi degli amici da quelli dei colleghi di lavoro o dei compagni della squadra di calcio. Ogni nominativo può appartenere a non più di tre gruppi, mentre i gruppi possono essere virtualmente infiniti. Ogni gruppo è identificato da tre lettere.

Per inserire un nominativo basta selezionare la scheda "new" e attivare i campi d'interesse con il mouse. Il bordo del campo attivato diventa più scuro e al suo interno compare il cursore. In ogni campo non può risiedere più di un certo numero di caratteri, variabile con il tipo di campo. I caratteri che superano il numero massimo vengono ignorati.

Una volta inseriti i nominativi, *geoDex* offre diverse soluzioni per effettuare una ricerca. È possibile per esempio sfogliare l'intero schedario agendo con il mouse sull'apposita icona che rappresenta la piega della pagina, o servendosi dei tasti cursore. Oppure selezionare uno dei caratteri alfabetici che separano le schede l'una dall'altra. In questo caso *geoDex* visualizza il primo nominativo individuato dalla lettera prescelta (in ordine alfabetico) o dall'asterisco. Tutte le schede in cui il campo del cognome non inizia con una lettera alfabetica vengono individuate dall'asterisco.

Oltre a questo tipo di ricerca "alfabetica", *geoDex* mette a disposizione anche una ricerca "personalizzata" tramite l'icona Search. In questo caso compare un box di dialogo che chiede all'utente d'inserire il cognome del nominativo che si desidera individuare. Se si ricorda solo parte di un cognome, si possono impiegare i due caratteri jolly (? e *) per rendere la ricerca più flessibile. Selezionando le icone di cambio pagina o agendo sui tasti cursore è poi possibile scorrere tutti i nominativi che rispondono alla chiave di ricerca.

Se si desidera restare nell'ambito di uno dei possibili gruppi, è sufficiente selezionare l'icona Group e indicare il gruppo prescelto. Fino a quando non viene selezionato

un altro gruppo, o non si disabilita la suddivisione in gruppi, tutte le funzioni agiscono solo sul gruppo selezionato (la stampa, la ricerca...).

Un altro aspetto interessante di *geoDex* è la stampa dei nominativi, che può essere limitata e selettiva. Se è stato selezionato un gruppo, la stampa viene effettuata solo per i nominativi che appartengono al gruppo. È possibile richiedere la stampa di tutti i numeri telefonici dei nominativi presenti nello schedario, la stampa di tutti gli indirizzi in forma di etichette (spaziate fra loro di un pollice o di un pollice e mezzo, in modo da adattarsi al tipo di etichette impiegate). Oppure è possibile richiedere la stampa integrale di tutte le schede. La stampa viene sempre effettuata in caratteri ASCII, contrariamente a quanto avviene di solito in ambiente *GEOS*.

Si può anche richiedere a *geoDex* di effettuare la chiamata telefonica dei numeri presenti nelle schede selezionate. Il manuale specifica che si dovrebbe collegare alla user port un modem Commodore autodial.

L'ultima caratteristica offerta da *geoDex* consiste nella generazione di un apposito file di testo *geoWrite* compatibile, che può essere impiegato con *geoMerge*. Questo raccoglie integralmente tutti i nominativi presenti nello schedario, o eventualmente solo quelli d'un particolare gruppo. Il formato per elencare i campi di ogni scheda è quello riconosciuto da *geoMerge* per stampare tante copie di un documento quanti sono i nominativi elencati: *geoMerge* accede a questo file per sostituire i campi presenti nel documento con il contenuto delle schede.

geoMerge richiede all'utente di sostituire un simbolo

geoMerge

Quest'applicazione consente di stampare più copie di uno stesso documento variandone gli attributi secondo le proprie esigenze. In pratica, l'utente prepara un testo con *geoWrite* e inserisce un particolare simbolo ogni qual volta desidera indicare una parte di testo che sarà diversa in ogni copia del documento. Quando *geoMerge* inizia a stampare le copie, ogni volta che incontra un simbolo richiede all'utente d'indicare la frase con cui dovrà sostituire il simbolo. Per esempio, nel caso di una lettera, ogni copia dello stesso testo può essere diversa per quanto riguarda destinatario, indirizzo, forma del saluto...

In questo modo l'utente scrive un testo solo, avendo cura d'indicare, tramite simboli come <<cognome>>, <<nome>> e così via, tutte le parti del testo che variano di volta in volta. Provvederà poi *geoMerge* a interrogare l'utente sui simboli di ogni copia che gli viene ordinata.

I simboli possono anche creare strutture condizionali: in base al valore che assume un simbolo nella copia corrente, *geoMerge* stampa una parte di testo o un'altra. Per esempio, se il destinatario può essere sia uomo che donna, il testo potrebbe comporsi di una parte fissa e di due parti che si escludono a vicenda. In questo caso si inserisce un simbolo condizionale come si può vedere nell'esempio che segue.

Ciao <<IF sesso = M>> mio caro amico <<ELSE>> mia cara amica <<ENDIF>>, come stai?

Se l'utente quando viene interrogato sul simbolo sesso risponde con una M, *geoMerge* stamperà la prima possibilità, e viceversa.

Le informazioni sui simboli che *geoMerge* inserisce di volta in volta all'interno delle copie del testo possono provenire, com'è stato spiegato, direttamente dall'utente, ma anche da un apposito file di testo. Questo file, generato con *geoWrite*, deve riportare, seguendo un'opportuna sintassi, tutte le informazioni che andranno a inserirsi in ogni copia del testo. In questo modo il processo di personalizzazione delle copie, il numero e la stampa, diventano tutte operazioni automatiche. Ovviamente il file deve riportare le informazioni appropriate per ciascuno dei simboli presenti nel testo.

Il file contenente le informazioni da sostituire in ogni copia può essere preparato anche da *geoDex*, come abbiamo già visto.

Questa applicazione si rivela molto utile per gestire in maniera più intelligente e funzionale la corrispondenza, e in generale, la stampa di testi standard parzialmente personalizzati.

deskTop V1.3

All'interno del pacchetto *geoDex* è compresa la versione 1.3 di *deskTop*, che sarà molto utile a chi non ne è ancora in possesso. Vediamo le novità che introduce rispetto alla versione precedente.

Per prima cosa, il nome del driver di stampa prescelto viene ora visualizzato sotto il disegno della stampante.

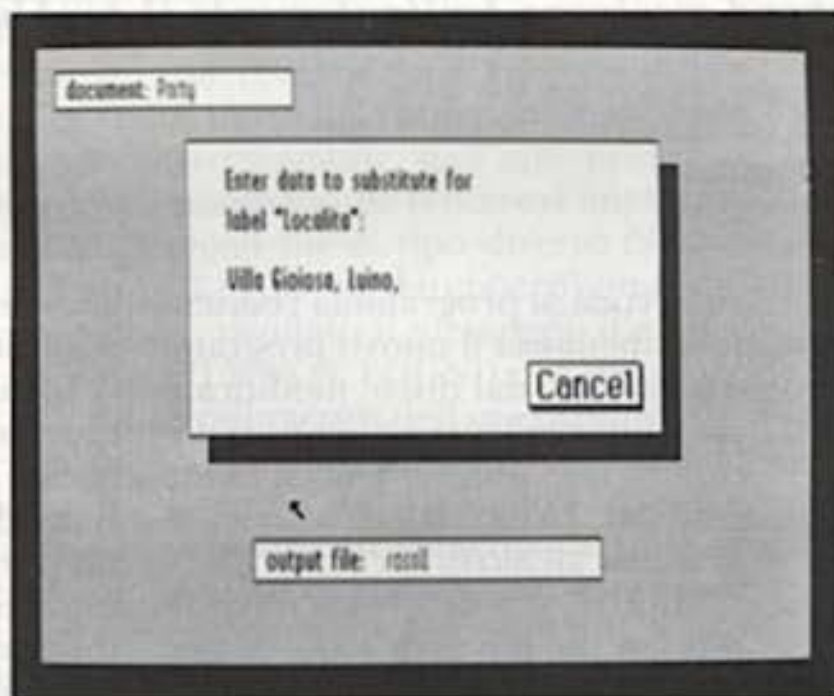
Vengono poi riconosciute alcune combinazioni di tasti che permettono di attivare particolari funzioni senza ricorrere ai menu e alle icone.

Con questa versione vi sono inoltre alcune operazioni che sui dischi "boot" (o di caricamento) e sui dischi "master" (contenenti le applicazioni originali), non sono permesse. Sono le operazioni di formattazione del disco e di cancellazione dei file.

Oltre alla nuova versione di *deskTop*, sono riportati in questo pacchetto alcuni nuovi driver di input e di stampa. In particolare, i driver di input permettono d'impiegare il mouse proporzionale 1351 della Commodore, la tavoletta grafica Koala, e la penna ottica.

Conclusioni

Questo pacchetto oltre a riunire un discreto numero di applicazioni e desk accessory, è interessante soprattutto perché può essere impiegato sia nel modo 40 colonne sia nel modo 80 colonne con GEOS 128. Il manuale è sufficientemente chiaro e riporta diversi esempi, soprattutto per impiegare *geoMerge*.

Nella serie delle applicazioni *GEOS* compatibili prodotte dalla Berkeley, *Graphics Grabber*, *Icon Editor*, *geoDex* e *geoMerge* sono applicazioni di minor peso, ma che all'occorrenza possono dimostrarsi preziose.

Suddividiamo in moduli i nostri programmi Basic

Lavorando con il Basic nasce frequentemente l'esigenza di inserire nel programma routine o linee realizzate a parte. Ma digitandole si perde tempo e si accumulano errori. Merge e Append risolvono brillantemente il problema

di Gianluca Pellegatta

Dopo aver presentato due routine di *Merge* e *Append* per il C-128 sul numero 5/87, le riproponiamo oggi nella versione per il C-64, computer che vanta una fascia di utenti ancora più estesa.

Può anche darsi che qualcuno possieda già queste due routine, ma nella maggior parte dei casi sono incluse nei famosi TOOL KIT (programmi che espandono il Basic di sistema). In genere contengono decine e decine di nuovi comandi, di cui alcuni davvero inutili, sono molto lunghi da caricare, difficili da usare e inoltre molto spesso creano problemi di compatibilità con il Basic residente. Le cose cambiano radicalmente disponendo di singole routine per ogni comando, perché queste si caricano in pochi secondi, sono facili da usare e non impegnano la memoria di sistema con codici inutili allo svolgimento dell'operazione desiderata.

Per questi motivi sono state realizzate queste due semplici ma efficaci routine. Vediamone il funzionamento. In pratica entrambi i comandi servono per unire tra loro due o più programmi Basic.

APPEND si limita ad aggiungere in coda al programma residente in memoria il nuovo programma caricato dal disco, modificando semplicemente i puntatori d'inizio e fine programma: si limita cioè ad "APPENDere" un programma all'altro.

MERGE, invece, svolge un compito un po' più elaborato. Legge dal file programma residente su disco ogni singola linea Basic e la trasferisce integralmente nel buffer di tastiera, simulandone così l'inserimento da tastiera. Quando arriva alla linea dopo la quale si vuole inserire il nuovo testo Basic, interrompe la lettura del file programma e inizia a inserire le linee del programma da includere. In seguito riprende a inserire le linee del file programma per terminare l'operazione MERGE. In questo modo i due programmi vengono amalgamati evitando ripetizioni di linee Basic fra loro comuni.

Risulta subito evidente che APPEND è senz'altro più veloce di MERGE, soprattutto per programmi lunghi, anche se in certi casi è meno efficace; comunque per ulteriori delucidazioni riguardo al funzionamento delle due routine si veda il disassemblato delle routine stesse, ampiamente commentato, in grado di chiarire ogni dubbio.

Da un punto di vista pratico, quando potrebbero rivelarsi utili questi due nuovi comandi per il Basic del C-64? Gli esempi possono essere numerosi. A molti sarà capitato più volte di realizzare programmi piuttosto lunghi o comunque complessi con altre persone, con la possibilità di suddividere così il lavoro in più moduli da svilupparsi ognuno per conto proprio. In questo caso, una volta realizzati i singoli moduli e verificata la loro funzionalità, si presenta il problema di riunirli fra loro per ottenere l'intero programma. Una routine di APPEND o di MERGE, a seconda dei casi, è la soluzione migliore, soprattutto in termini di praticità e tempo. Si tratta di impiegare APPEND se i diversi moduli del programma hanno i numeri di linea già ordinati e crescenti, oppure MERGE in caso contrario, perché quest'ultima routine provvede automaticamente a riordinare i numeri di linea.

Un altro esempio potrebbe essere il caso in cui il programmatore desidera approntare una propria libreria di routine Basic in grado di far fronte alle nec

più frequenti. In questo caso diventa molto comodo servirsi dei comandi MERGE e APPEND nella realizzazione dei programmi, caricando da disco le routine necessarie. L'ulteriore vantaggio nella modularità dei programmi è la sicurezza che non interverranno errori di digitazione e che i moduli sono perfettamente funzionanti perché già collaudati uno per uno, prima della fusione in un unico programma.

L'uso dei programmi

Vediamo ora la procedura necessaria per installare questi due comandi sul C-64. Per prima cosa bisogna digitare il programma relativo al comando di cui ci si vuole servire. Si può indifferentemente impiegare il programma Basic di caricamento, oppure, disponendo di un monitor o di un compilatore Assembly, digitare il programma direttamente in linguaggio macchina.

Una volta che la routine è stata digitata e salvata su disco, le operazioni da fare per attivare il comando sono le seguenti: si carica la routine prescelta, facendo attenzione però che, se non è in Basic, si rende necessario effettuare un caricamento rilocato (aggiungendo cioè ",1" al comando LOAD del Basic). Se il programma d'installazione è in Basic, è sufficiente impartire il comando RUN seguito dal comando NEW. Se invece la routine è interamente in linguaggio macchina è sufficiente impartire il comando NEW.

A questo punto la routine è in memoria, pronta per essere impiegata. Si comincia caricando il primo programma, dopodiché si digita:

```
SYS 49152,"filename",8
```

dove "filename" rappresenta il nome del secondo programma da fondere o appendere con il primo. Se non sono stati commessi errori nel nome del file o nella sintassi del comando, nel qual caso verrebbe visualizzato un opportuno messaggio d'errore, dopo un'attesa dipendente dalla routine scelta e dalla lunghezza dei programmi appare sullo schermo un messaggio per segnalare che l'operazione è stata completata con successo; a questo punto in memoria risiede un programma unico ottenuto unendo i singoli moduli, e che può essere salvato su disco.

Particolari tecnici

Prima di concludere occorre qualche precisazione tecnica sui comandi MERGE e APPEND. Per come sono stati strutturati, questi due comandi operano solo su file di tipo PRG contenenti dati Basic riconoscibili dall'interprete; è quindi inutile tentare d'impiegarli con file di tipo diverso (SEQ o REL) perché darebbero come unico risultato il messaggio d'errore FILE NOT FOUND per segnalare il fallimento dell'operazione.

Le routine che realizzano le funzioni MERGE e APPEND occupano rispettivamente la zona di memoria da $C000 (decimale 49152) a $C10C (49420) e da $C000 a $C084 (49284).

La zona di memoria prescelta per allocare le routine è completamente al di fuori della portata dei programmi Basic per cui non dovrebbero presentarsi problemi di compatibilità con nessun tipo di programma, a patto però che le aree di memoria appena citate non vengano alterate da istruzioni POKE.

Per quanto riguarda un eventuale rilocamento delle routine allo scopo di soddisfare particolari esigenze, è necessario distinguere due casi. Se le routine sono state create con un compilatore Assembly, è sufficiente intervenire sul file sorgente cambiando l'indirizzo al quale il file verrà rilocato. Se invece le routine sono state realizzate con un monitor, il compito si presenta più arduo e bisogna modificare tutti gli indirizzi assoluti che fanno riferimento a locazioni del programma. Per quanto riguarda i programmi Basic, rilocare le routine significa riscriverle con un monitor e trasformarle in DATA eseguibili da Basic: un lavoro complesso e decisamente sconsigliabile.

Vediamo infine l'elenco delle routine del Kernel e dell'interprete Basic che vengono impiegate da MERGE e APPEND.

ERROR = $A437 Stampa il messaggio d'errore il cui codice è contenuto nel registro x.

READY = $A474 Stampa sullo schermo il messaggio "READY." ed esce dal programma.

INSLIN = $A4A4 Trasferisce i dati in ingresso dal buffer di input al programma; il numero di linea è contenuto nelle locazioni $14 e $15 e la lunghezza della linea in Y.

LNKPRG = $A533 Provvede al concatenamento dell'attuale linea di programma con le precedenti, ristabilendone i puntatori.

STROUT = $AB1E Stampa la stringa puntata dall'accumulatore (byte basso) e dal registro Y (byte alto) finché non trova un byte a 0 che ne indica la fine.

SLPARA = $E1D4 Preleva i parametri dal testo Basic per le routine di SAVE, LOAD e VERIFY.

PRTFIL = $F5C1 Stampa il nome del file o meglio dell'ultima stringa usata tra virgolette.

LOAD = $FFD5 Carica o verifica (da nastro oppure da disco) a seconda del valore contenuto nell'accumulatore: 0=load, 1=verify.

OPEN = $FFC0 Apre un file logico per le procedure di I/O.

CHKIN = $FFC6 Apre un canale di input per prelevare i dati da una periferica.

CLOSE = $FFC3 Chiude il file logico il cui numero è contenuto nell'accumulatore.

CLRCHN= $FFCC Azzera tutti i canali di I/O aperti, reimpostandoli ai loro rispettivi valori di default.

CHRIN = $FFCF Preleva un carattere da un canale di input e lo memorizza nell'accumulatore

```
;##########################################################################################
;###                                                                                    ###
;###################################  MERGE per C-64  ####################################
;###                                                                                    ###
;##########################################################################################
;
;dichiarazione dei principali simboli e dei nomi che faranno riferimento
;alle routine e alle locazioni utilizzate nel programma
;
;
       la       =$00B8   ;numero del file logico attivo
       sa       =$00B9   ;indirizzo secondario corrente
       fa       =$00BA   ;numero di device attivo
       status   =$0090   ;variabile di stato delle operazioni di I/O del Kernel
                         ;corrisponde alla variabile "st" del Basic
       buf      =$0200   ;indirizzo base del buffer di input del sistema
       tabfil   =$0258   ;indirizzo base della tavola del Kernel
                         ;dei numeri di file logici attivi
       lstx     =$00C5   ;tasto attualmente premuto
       count    =$000B   ;puntatore del buffer di input
       linuml   =$0014   ;byte basso & byte alto
       linumh   =$0015   ;del numero di linea attuale
       ldtnd    =$0098   ;numero dei file aperti,
       imainl   =$0302   ;byte basso & alto dell'indirizzo
       imainh   =$0303   ;del vettore di "warm start"
       iclall   =$032C   ;byte basso & alto dell'indirizzo
       iclalh   =$032D   ;del vettore di clall (chiude tutti i file)
       filnum   =$C0F9   ;numero del file usato per il merge
       flag     =$C0FA   ;variabile di comodo
       error    =$A437   ;routine di stampa messaggi di errore
       ready    =$A474   ;stampa "ready"
       inslin   =$A4A4   ;inserimento linee Basic
       strout   =$AB1E   ;routine di stampa stringhe
       slpara   =$E1D4   ;preleva i parametri dal testo Basic
                         ;per le routine di save, load e verify
       prtfil   =$F5C1   ;stampa il nome del file
       open     =$FFC0   ;apre un file logico
       chkin    =$FFC6   ;apre un canale di input
       close    =$FFC3   ;chiude un file logico
       clrchn   =$FFCC   ;azzera tutti i canali aperti di I/O
       chrin    =$FFCF   ;preleva un carattere da un canale di input
;
;##########################################################################################
;   inizio routine di merge
;##########################################################################################
;
;##########################################################################################
;   inizializzazione canale di input
;##########################################################################################
;
;
       *=$C000           ;origine del programma
       lda   #$00        ;azzera filnum e flag
       sta   filnum
       sta   flag
       jsr   slpara      ;controlla che il numero di device
       lda   fa
       cmp   #$08        ;sia uguale a "8"
       beq   ok
       ldx   #$09        ;altrimenti stampa errore
       jsr   error
ok     lda   #$02        ;stampa messaggio di routine in corso (merging)
       ldy   #$C1
       jsr   strout
       jsr   prtfil      ;stampa nome file
```

```
        lda     #$0E
        jsr     ctrfil      ;predispone il nome del file
        sta     la
        sta     filnum
        lda     #$00        ;indirizzo secondario = 0
        sta     sa
        jsr     open        ;apre il file logico
        ldx     filnum
        jsr     chkin       ;apre il canale di input
        lda     #$E1        ;modifica il vettore di "clall" che chiuderebbe il file
                            ;e lo indirizza a un'istruzione di ritorno da subroutine (rts)
        sta     iclall
        lda     #$C0
        sta     iclalh
        lda     #$51        ;modifica il vettore di "warm start" e lo indirizza alla
                            ;routine di lettura e inserimento linee
        sta     imainl
        lda     #$C0
        sta     imainh
;
;
;###########################################################################################
;  lettura e memorizzazione delle linee Basic
;###########################################################################################
;
;
        jsr     chrin       ;legge i primi 2 byte del nuovo programma
        jsr     chrin       ;che corrispondono all'indirizzo iniziale
                            ;di quest'ultimo e che quindi non interessano il programma residente
neulinjsr       chrin       ;preleva i due byte corrispondenti all'indirizzo
                            ;della linea Basic che si sta leggendo
        sta     linuml
        jsr     chrin       ;e li memorizza temporaneamente in "linuml" & "linumh"
        sta     linumh
        ora     linuml
        beq     exit        ;se sono entrambi a zero significa fine programma
        lda     status      ;controlla status
        bne     exit        ;se ci sono errori di I/O interrompe il merge
        jsr     chrin       ;preleva i due byte del numero di linea
        sta     linuml      ;li memorizza in "linuml" & "linumh"
        jsr     chrin       ;per poter poi essere trattati dalla routine "inslin"
        sta     linumh
        ldy     #$00
loop    jsr     chrin       ;legge i caratteri della linea Basic
        sta     buf,y       ;e li memorizza nel buffer
        ldx     lstx        ;controlla che non venga premuto "run/stop"
        cpx     #$3F
        bne     goon        ;altrimenti interrompe il merge stampando l'opportuno
                            ;messaggio di errore
        stx     flag
        jmp     exit
goon    iny
        cmp     #$00        ;continua fino a quando non incontra uno zero
                            ;che sta a significare la fine della linea Basic corrente
        bne     loop        ;altrimenti esce dal loop
        tya
        clc
        adc     #$04
        sta     count       ;memorizza in "count" il numero di caratteri della linea
                            ;appena letta ("count" corrisponde al puntatore del buffer)
        lda     status      ;controlla ancora che non ci siano errori di I/O
        bne     exit
        ldy     count       ;memorizza "count" in y
        jmp     inslin      ;quindi chiama la routine di inserimento nuova linea
                            ;che provvede automaticamente a prelevare il contenuto del buffer e a
                            ;riporlo in memoria al giusto indirizzo e con gli opportuni puntatori
        jmp     neulin      ;salta a lettura nuova linea
;
;
```

```
;
;****************************************************************************
;   controllo errori, chiusura canali di I/O
;****************************************************************************
;
;
exit   lda      #$83      ;ripristino vettore di "warm start"
       sta      imainl
       lda      #$A4
       sta      imainh
       lda      #$2F      ;ripristino vettore di "clall"
       sta      iclall
       lda      #$F3
       sta      iclalh
       lda      filnum
       jsr      close     ;chiusura file logico
       jsr      clrchn    ;azzeramento canale di input
       lda      flag      ;controllo tipo di errore
       cmp      #$3F
       bne      exit2
       lda      #$00
       sta      flag
       ldx      #$1E      ;errore di interruzione
       jsr      error     ;stampa messaggio
       jmp      ready     ;stampa ready
exit2  lda      status
       cmp      #$42
       bne      exit3
       ldx      #$04      ;errore di file non trovato
       jsr      error     ;stampa messaggio
       jmp      ready     ;stampa ready
exit3  lda      #$FB      ;nessun errore
       ldy      #$C0
       jsr      strout    ;stampa fine merge
       jmp      ready     ;stampa ready
       rts
;
;
;****************************************************************************
;   controllo su file eventualmente gia' aperti
;****************************************************************************
;
;
ctrfil ldx      ldtnd     ;carica numero di file aperti
       cpx      #$00
       beq      out       ;se zero esce
loop2  cmp      tabfil,x  ;altrimenti ricerca il primo disponibile
       bne      repeat    ;a partire da $0E (vedere inizializzazione)
       sec
       sbc      #$01
       jmp      ctrfil
repeat dex                ;decrementa l'indice della tavola file
       beq      out       ;se uguale a zero esce
       bne      loop2     ;altrimenti ritorna a "loop2"
out    rts
;
;
;****************************************************************************
;   riserva locazioni per "numfil" & "flag"
;****************************************************************************
;
;
       .byte    00        ;"numfil"
       .byte    00        ;"flag"
;
;
;****************************************************************************
;   converte il messaggio "end of merging" nei corrispettivi
;   codici ASCII per la stampa su schermo
;****************************************************************************
```

```
;
;
        .byte     17              ;codice ascii di [shift + return]
        .byte     "end of merging ",0

        .byte     00
;
        .end                      ;fine programma
```

```
;#################################################################################
;###                                                                           ###
;######################################    APPEND  per  C-64    #################
;###                                                                           ###
;#################################################################################
;
;
;#################################################################################
;   dichiarazione nomi e simboli
;#################################################################################
;
        verck     =$000A          ;flag di sistema: 0=load; 1=verify
        vrtabl    =$002D          ;byte basso e byte alto del puntatore dell'inizio
                                  ;dell'area delle variabili Basic, cioe' la fine dell'area
                                  ;occupata dal testo Basic
        vrtabh    =$002E
        artabl    =$002F          ;byte alto e byte basso del puntatore dell'inizio
                                  ;dell'area occupata dagli array
        artabh    =$0030
        strenl    =$0031          ;fine area array+1, inizio area stringhe
        strenh    =$0032
        status    =$0090          ;variabile di stato
        fa        =$00BA          ;numero di device corrente
        lstx      =$00C5          ;tasto attualmente premuto
        point1    =$00FC          ;RAM libera in pagina zero utilizzata dal programma
        point2    =$00FD          ;per memorizzare i puntatori delle varie linee Basic
        help      =$00FE
        error     =$A437          ;stampa il messaggio di errore il cui codice
                                  ;e' contenuto nel registro x
        lnkprg    =$A533          ;routine dell'interprete Basic: provvede al
                                  ;concatenamento dell'attuale linea di programma con le precedenti,
                                  ;ristabilendone i puntatori
        slpara    =$E1D4          ;preleva i parametri dal testo Basic per le routine
                                  ;di save, load e verify
        load      =$FFD5          ;carica o verifica da nastro oppure da disco.
                                  ;0=load, 1=verify
;
;
;#################################################################################
;         inizializzazione
;#################################################################################
;
;
        *=$C000                   ;origine del programma

        lda       #$00
        sta       verck           ;predispone al caricamento del programma da "appendere"
        jsr       slpara
        lda       $FA
        cmp       #$08            ;controlla che il numero di device sia corretto
        beq       ok
        ldx       #$09            ;altrimenti stampa errore
        jsr       error
ok      lda       vrtabl          ;ridimensiona il puntatore di inizio area variabili
```

```
        sec
        sbc     #$02        ;in fondo al programma residente in memoria
        tax
        lda     vrtabh      ;per poter caricare il secondo
        sbc     #$00
        tay
        lda     #$00
        jsr     load        ;carica il nuovo programma
        lda     status      ;ci sono errori?
        beq     noerr       ;no, prosegui
        jsr     err         ;si', salta a gestione errori
noerr   jsr     lnkprg      ;non ci sono errori quindi richiamo la routine "lnkprg"
                            ;per il concatenamento delle linee del nuovo programma
;
;
;###########################################################################
;  loop di scansione puntatori
;   per ricerca fine programma
;###########################################################################
;
;
        lda     vrtabl      ;decrementa di due il puntatore vrtab
        ldx     vrtabh      ;per cancellare i due zeri di fine programma residente
        sec
        sbc     #$02
        sta     point1      ;e memorizza il risultato in point1 & point2
        txa
        sbc     #$00
        sta     point2
loop    ldy     #$00        ;inizio loop scansione puntatori
        lda     (point1),y
        bne     nxtlin      ;se sono entrambi uguali a zero significa che si e' giunti
                            ;alla fine del nuovo programma e quindi si esce dal ciclo andando
                            ;ad aggiornare i puntatori di inizio variabili, array e stringhe
        iny
        lda     (point1),y
        beq     exit
;
;
;###########################################################################
;  aggiornamento puntatore prossima linea Basic
;###########################################################################
;
;
nxtlin  jsr     err2        ;controllo errori
        ldy     #$00
        lda     (point1),y  ;legge il puntatore corrente (byte basso)
        sta     help        ;e lo memorizza temporaneamente in "help"
        iny
        lda     (point1),y  ;legge il puntatore corrente (byte alto)
        sta     point2      ;e lo memorizza in "point2"
        lda     help        ;recupera "help" e lo memorizza in "point1"
        sta     point1
        jmp     loop        ;salta a loop per una nuova linea
;
;
;###########################################################################
;  aggiornamento puntatori per fine programma
;###########################################################################
;
;
exit    lda     point1      ;carica point1 che contiene il puntatore (byte basso)
                            ;di fine programma (residente + nuovo), lo incrementa di due per lo
                            ;stesso motivo visto prima e lo memorizza
        clc
        adc     #$02
        sta     vrtabl      ;in "vrtabl", "artabl" e "strenl"
        sta     artabl
        sta     strenl
```

```
        lda     point2      ;carica point2, cioe' il byte alto del puntatore di fine
                            ;programma e, tenendo conto di un eventuale riporto della somma
                            ;precedente con point1, lo memorizza
        adc     #$00
        sta     vrtabh      ;in "vrtabh", "artabh" e "strenh"
        sta     artabh
        sta     strenh
        rts
;
;
;!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
;   subroutine di gestione errori
;!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
;
;
err     cmp     #$42        ;controllo su errore di "file not found"
        bne     err2
        ldx     #$04
        jsr     error
err2    lda     lstx        ;controllo su eventuale pressione di "run/stop"
        cmp     #$3F
        bne     return
        ldx     #$1E
        jsr     error
return  rts                 ;ritorno al ciclo in assenza di errori
;
        .end                ;fine programma
;
```

Listato: *Merge C-64*

```
10 REM ------------------------------
20 REM ----                     ----
30 REM ----    MERGE PER C= 64  ----
40 REM ----                     ----
50 REM ----  (C) 1987  BY :     ----
60 REM ----                     ----
70 REM ---- PELLEGATTA GIANLUCA ----
80 REM ----                     ----
90 REM ------------------------------
100 :
110 PRINT"      ROUTINE DI MERG
    E PER COMMODORE 64"
120 PRINT"   (C) 1987 BY  PEL
    LEGATTA GIANLUCA"
130 PRINT"ATTENDERE QUALCHE
     ISTANTE:"
140 PRINT"LETTURA DATI IN CORSO
    "
150 REM *** CICLO DI LETTURA **
    *
160 IND=49152
170 READ C:IF C=-1 THEN 200
180 POKE IND,C:IND=IND+1:CK=CK+
    C
190 GOTO 170
200 IF CK=33046 THEN 230
210 PRINT"ATTENZIONE: ERRORE
    NELLE LINEE DATA,"
220 PRINT"          RICONTROL
    LATE IL LISTATO":GOTO 998
230 PRINT"OK, TUTTO A POSTO"
240 PRINT"DATE IL NEW, CARICATE
    IL PRIMO PROGRAMMA QUINDI
    DIGITATE:"
250 PRINT"SYS49152"+CHR$(34)+"
    FILENAME"+CHR$(34)+",8"
997 :
998 END
999 :
1000 DATA 169, 000, 141, 249, 1
     92, 141, 250, 192,  32, 21
     2, 225, 165, 186, 201
1005 :
1010 DATA   8, 240,   5, 162,
      9,  32,  55, 164, 169,
     2, 160, 193,  32,  30
1015 :
1020 DATA 171,  32, 193, 245, 1
     69,  14,  32, 226, 192, 13
     3, 184, 141, 249, 192
1025 :
1030 DATA 169, 000, 133, 185,
     32, 192, 255, 174, 249, 19
     2,  32, 198, 255, 169
1035 :
1040 DATA 225, 141,  44,   3, 1
     69, 192, 141,  45,   3, 16
     9,  81, 141,   2,   3
1045 :
1050 DATA 169, 192, 141,   3,
      3,  32, 207, 255,  32, 20
     7, 255,  32, 207, 255
1055 :
1060 DATA 133,  20,  32, 207, 2
     55, 133,  21,   5,  20, 24
     0,  57, 165, 144, 208
1065 :
1070 DATA  53,  32, 207, 255, 1
     33,  20,  32, 207, 255, 13
     3,  21, 160, 000,  32
1075 :
1080 DATA 207, 255, 153, 000,
     2, 166, 197, 224,  63, 20
     8,   6, 142, 250, 192
1085 :
1090 DATA  76, 152, 192, 200, 2
     01, 000, 208, 233, 152,  2
     4, 105,   4, 133,  11
1095 :
1100 DATA 165, 144, 208,   8, 1
     64,  11,  76, 164, 164,  7
     6,  81, 192, 169, 131
1105 :
1110 DATA 141,   2,   3, 169, 1
     64, 141,   3,   3, 169,  4
     7, 141,  44,   3, 169
1115 :
1120 DATA 243, 141,  45,   3, 1
     73, 249, 192,  32, 195, 25
     5,  32, 204, 255, 173
1125 :
1130 DATA 250, 192, 201,  63, 2
     08,  13, 169, 000, 141, 25
     0, 192, 162,  30,  32
1135 :
1140 DATA  55, 164,  76, 116, 1
     64, 165, 144, 201,  66, 20
     8,   8, 162,   4,  32
1145 :
1150 DATA  55, 164,  76, 116, 1
     64, 169, 251, 160, 192,  3
     2,  30, 171,  76, 116
1155 :
1160 DATA 164,  96, 166, 152, 2
     24, 000, 240,  16, 221,  8
     8,   2, 208,   6,  56
1165 :
1170 DATA 233,   1,  76, 226, 1
     92, 202, 240,   2, 208, 24
     0,  96, 000, 000,  17
1175 :
1180 DATA  69,  78,  68,  32,
     79,  70,  32,  77,  69,  8
     2,  71,  73,  78,  71
1185 :
1190 DATA  32, 000, 000, -1
1195 :
1200 END
```

Listato: *Append C-64*

```
10 REM -----------------------------
20 REM -------
30 REM -------    APPEND PER C= 64
40 REM -------
50 REM -------    (C) 1987  BY :
60 REM -------
70 REM ------- PELLEGATTA GIANLUCA
80 REM -------
90 REM -----------------------------
100 :
110 PRINT"     ROUTINE DI APPE
    ND PER COMMODORE 64"
120 PRINT"     (C) 1987 BY  PEL
    LEGATTA GIANLUCA"
130 PRINT"    ATTENDERE QUALCHE
    ISTANTE:"
140 PRINT"LETTURA DATI IN CORSO
    "
150 REM *** CICLO DI LETTURA **
    *
160 IND=49152
170 READ C:IF C=-1 THEN 200
180 POKE IND,C:IND=IND+1:CK=CK+
    C
190 GOTO 170
200 IF CK=16634 THEN 230
210 PRINT"  ATTENZIONE  : ERRORE
    NELLE LINEE DATA,"
220 PRINT"              RICONTROL
    LATE IL LISTATO":GOTO 998
230 PRINT" OK, TUTTO A POSTO"
240 PRINT"DATE IL NEW, CARICATE
    IL PRIMO PROGRAMMA QUINDI
    DIGITATE:"
250 PRINT" SYS49152"+CHR$(34)+"
    FILENAME"+CHR$(34)+",8"
997 :
998 END
999 :
1000 DATA 169, 000, 133,  10,
     32, 212, 225, 165, 186, 20
     1,   8, 240,   5, 162
1005 :
1010 DATA   9,  32,  55, 164, 1
     65,  45,  56, 233,   2, 17
     0, 165,  46, 233, 000
1015 :
1020 DATA 168, 169, 000,  32, 2
     13, 255, 165, 144, 240,
     3,  32, 112, 192,  32
1025 :
1030 DATA  51, 165, 165,  45, 1
     66,  46,  56, 233,   2, 13
     3, 252, 138, 233, 000
1035 :
1040 DATA 133, 253, 160, 000, 1
     77, 252, 208,   5, 200, 17
     7, 252, 240,  21,  32
1045 :
1050 DATA 121, 192, 160, 000, 1
     77, 252, 133, 254, 200, 17
     7, 252, 133, 253, 165
1055 :
1060 DATA 254, 133, 252,  76,
     58, 192, 165, 252,  24, 10
     5,   2, 133,  45, 133
1065 :
1070 DATA  47, 133,  49, 165, 2
     53, 105, 000, 133,  46, 13
     3,  48, 133,  50,  96
1075 :
1080 DATA 201,  66, 208,   5, 1
     62,   4,  32,  55, 164, 16
     5, 197, 201,  63, 208
1085 :
1090 DATA   5, 162,  30,  32,
     55, 164,  96, -1
```

L'ARCHITETTURA DEL SISTEMA OPERATIVO GEOS: SECONDA PARTE

GEOS: ANATOMIA DI UN SISTEMA OPERATIVO

GEOS dispone di un "cervello" e di un "sistema nervoso" che insieme gestiscono gli eventi in multitasking. Questi due "enti" mantengono il completo controllo delle interazioni fra sistema, utente e hardware

di Luca Giachino

Questo articolo si inserisce in un ciclo dedicato al sistema operativo GEOS. *Per una lettura agevole è necessario conoscere gli argomenti trattati nella precedente puntata.*

Eccoci nuovamente alle prese con il sistema operativo *GEOS*. Nella prima parte di questa serie sono stati illustrati alcuni argomenti che costituiscono la vera e propria ossatura su cui si regge la filosofia dell'intero sistema, fondamentali per comprenderne a fondo l'architettura.

Ricordiamo in due parole i temi principali illustrati nella prima parte. *GEOS* è un sistema operativo a gestione del flusso di eventi. L'evento è, in linea generale, qualcosa in grado di richiamare l'attenzione del sistema; per esempio, la selezione di un menu o di un'icona, l'introduzione di una stringa di testo, la selezione di un'area non convenzionale dello schermo (un'area diversa da un menu o da un'icona), lo scadere dell'intervallo stabilito per un processo temporizzato e la sua conseguente esecuzione...

Gli eventi vengono suddivisi in tre tipi fondamentali: gli eventi convenzionali (per esempio la selezione di un menu), gli eventi non convenzionali (per esempio la pressione di un tasto quando non sono in corso input di stringhe), e gli eventi temporizzati (per esempio l'aggiornamento di un orologio sullo schermo indipendentemente dalle operazioni in corso).

L'applicazione *GEOS* compatibile deve prevedere strutture di dati, le cosiddette tavole di definizione, che identifichino gli eventi di cui il sistema deve prendersi cura. Le informazioni in esse contenute non caratterizzano solo ogni evento, ma anche le interazioni fra l'uno e l'altro. Una volta che il sistema entra in possesso dell'intera struttura di definizione, procede alla gestione del flusso di eventi senza nessun aiuto da parte dell'applicazione.

A ogni evento "finale" (che provoca cioè un'operazione tipica dell'applicazione, e non del sistema) viene associata una routine di servizio: al verificarsi dell'evento il sistema accede alla tavola di definizione e manda in esecuzione la routine di servizio associata. Terminato il suo svolgimento, la routine cede il controllo al sistema che torna a gestire i flussi di eventi. Queste "routine di servizio" devono essere fornite dall'applicazione, anzi, in realtà ne costituiscono il cuore.

L'applicazione deve quindi svolgere fondamentalmente due compiti: inizializzare la gestione degli eventi comunicando al sistema le opportune tavole di definizione, e rendere disponibili le routine di servizio preposte allo svolgimento delle funzioni sottese agli eventi.

Questi, in sintesi, gli argomenti trattati nella prima parte. Dal momento che ricorreranno con molta frequenza nel corso di tutta la serie di articoli, è opportuno averli ben chiari. In questa seconda parte espanderemo le conoscenze di cui disponiamo analizzando le strutture all'interno del sistema che permettono la gestione degli eventi. Prima di proseguire è opportuno illustrare le convenzioni adottate per i nomi delle routine, delle variabili e delle costanti di sistema. Il linguaggio è simbolico e fa riferimento a quello impiegato nella *Guida Ufficiale alla Programmazione di GEOS* edita dalla IHT (la casa editrice di *Commodore Gazette*), dalla quale vengono attinte le informazioni riportate in questa serie di articoli. I nomi delle **routine** sono composti da una o più parole inglesi e ogni parola ha la prima lettera maiuscola e le successive minuscole (per esempio EnterDeskTop). **Le variabili**

ovvero i nomi simbolici assegnati alle locazioni di memoria (compresi vettori e buffer) destinate a contenere i dati, hanno invece la prima parola interamente minuscola (per esempio mouseFaultVec). Infine, le **costanti** vengono indicate interamente in maiuscolo (per esempio SET_MSE_ON).

Il cervello e il sistema nervoso

Il corpo umano, nella sua meravigliosa complessità, fornisce spesso significative analogie per descrivere organismi anche molto diversi da quelli umani. Inoltre, analizzare la struttura di un sistema mettendone in luce similitudini e corrispondenze con gli organi del nostro corpo – senza voler fare lezioni di anatomia – può aiutare a fissare i concetti fondamentali con immediatezza e semplicità.

Diremo quindi che *GEOS* possiede un "cervello" che elabora le informazioni pervenute sugli eventi, e un "sistema nervoso" che provvede a interagire con l'hardware della macchina per rendere disponibili i dati che caratterizzano gli eventi in corso. All'interno del Kernel di *GEOS* il "cervello" è MainLoop e il "sistema nervoso" è InterruptMain: sono due routine di elevata complessità che svolgono le funzioni principali, assolvendo compiti diversi ma strettamente complementari.

MainLoop

Questa è in assoluto la routine più importante del sistema. È costituita da un grande loop che continua a ripetersi. La sua funzione è controllare il flusso degli eventi, distinguendo quelli significativi da quelli non significativi. Per quanto riguarda, ad esempio, gli eventi convenzionali, MainLoop provvede a gestire l'intera struttura di un menu, e quando l'utente seleziona una voce che dovrebbe produrre un "evento finale", è in grado di accorgersene e di chiamare la corrispondente routine di servizio. Questa svolge la propria funzione e poi in genere restituisce il controllo a MainLoop, che riprende esattamente dal punto in cui si era interrotto.

Questo esempio illustra una tipica funzione svolta da MainLoop, ma non è sufficiente per metterne in luce la complessità: occorre approfondire la definizione di evento che avevamo introdotto nella prima parte.

Nella sua generalità, l'evento prodotto dall'utente (escludiamo volutamente quelli temporizzati, per il momento) è un qualsiasi tentativo di comunicazione con la macchina. Il semplice movimento del mouse o la pressione di un tasto (i due principali strumenti di comunicazione utente-sistema) sono eventi. InterruptMain li segnala e MainLoop procede ad analizzarli. Supponiamo di trovarci di fronte alla selezione della voce di un menu: MainLoop riceve da InterruptMain un "messaggio" del tipo "il pulsante del mouse è stato premuto". Di conseguenza provvede a controllare se la posizione corrente del mouse rientra in una delle aree definite "menu" o "icone". Se il mouse si trova in un'area diversa, l'evento registrato è di tipo non convenzionale. In questo caso, MainLoop chiama la routine puntata dal vettore di sistema otherPressVec.

Normalmente otherPressVec è azzerato per indicare che il sistema non prende in considerazione gli eventi non convenzionali (in ambiente *GEOS* se un vettore di sistema indica un indirizzo pari a $0000, non viene eseguita nessuna chiamata e si assume che il vettore non punti ad alcuna routine). Se l'applicazione desidera controllare anche un evento di questo tipo, ad esempio per spostare il cursore nella posizione corrente del mouse, deve memorizzare in otherPressVec l'indirizzo di una propria routine di gestione dell'evento non convenzionale.

Vediamo ora cosa accade quando il pulsante del mouse è stato premuto in una zona convenzionale. Se l'area è quella di un menu, MainLoop provvede ad aprire il sotto-menu, o, se non seguono altri livelli di menu, a chiamare la routine corrispondente. Se l'utente seleziona un'altra voce, MainLoop procede seguendo le disposizioni memorizzate nella tavola di definizione del menu. Senza scendere in particolari, possiamo dire semplicemente che se la voce selezionata dall'utente corrisponde a un evento finale (cioè non produce l'apertura di un altro livello di menu), MainLoop manda in esecuzione la routine di servizio associata: l'applicazione viene chiamata in causa solo al termine di una serie di eventi, tra i quali l'unico per essa significativo è l'ultimo. Le precedenti selezioni sono anch'esse eventi significativi, ma solo per il sistema, che provvede ad aprire i sotto-menu senza coinvolgere l'applicazione. Infatti, se l'utente esplora i vari livelli del menu, ma non seleziona nessuna voce finale, il sistema apre e chiude i diversi sotto-livelli mantenendo costantemente il controllo, e l'applicazione non viene invocata.

Questo esempio chiarisce quanto abbiamo voluto sottolineare in precedenza, cioè che *GEOS* è in grado di svolgere la maggior parte del lavoro in modo autonomo, coinvolgendo l'applicazione solo dopo un concatenamento di eventi. Inoltre, anche la distinzione fra eventi convenzionali e non convenzionali è ora più chiara: gli uni sono interamente gestiti dal sistema e sono composti da una serie di sotto-eventi completamente trasparenti all'applicazione, mentre gli altri, tramite i vettori di sistema, possono coinvolgere direttamente l'applicazione.

Un altro tipo di evento convenzionale è la selezione di un'icona. Come sempre MainLoop riceve da InterruptMain la segnalazione di una pressione del pulsante del mouse, e se verifica che il mouse si trova sopra un'icona, accede alla relativa tavola di defin[illegible]

per agire di conseguenza. Anticipiamo che le possibili operazioni che il sistema potrebbe compiere, oltre a chiamare la corrispondente routine di servizio, riguardano il modo in cui la selezione viene segnalata all'utente: l'icona può invertirsi e rimanere in questo stato, invertirsi solo per una frazione di secondo o non invertirsi affatto.

Per quanto riguarda gli eventi non convenzionali, un altro esempio è la pressione di un tasto quando non sono in corso input di stringa. MainLoop riceve da InterruptMain la segnalazione e provvede a chiamare la routine puntata dal vettore di sistema keyVector. Di solito anche questo vettore indica un indirizzo nullo, ma se l'applicazione desidera controllare i caratteri premuti dall'utente, vi memorizza l'indirizzo di un'apposita routine di gestione (deskTop agisce in questo modo per elaborare i caratteri premuti contemporaneamente al tasto Commodore).

Inoltre MainLoop è in grado di gestire gli eventi temporizzati, ovvero lo scadere di un tempo prefissato (espresso in chiamate di interrupt), e la successiva chiamata di un processo, chiamato per questo "processo temporizzato". L'applicazione che desidera attivare uno o più eventi temporizzati, deve fornire al sistema una tavola che li definisca. MainLoop accede ai dati in essa contenuti per sapere in quale "stato" deve attivare ogni evento (ci sono tre diversi "stati" possibili, che verranno illustrati in una delle prossime puntate), e per leggere l'intervallo temporale di ripetizione che caratterizza la periodicità del processo. In seguito, per ogni processo, aggiorna un timer con il valore dell'opportuno intervallo. InterruptMain provvede a decrementare il contenuto dei timer a ogni chiamata di interrupt. MainLoop è così in grado di accorgersi quando uno di questi diventa zero, e chiamare il corrispondente processo temporizzato. Questa breve carrellata sui processi temporizzati non è sufficiente per descriverne l'intera struttura (se ne riparlerà in una delle prossime puntate), ma ci ha permesso almeno di descrivere come MainLoop gestisce gli eventi temporizzati.

Ora disponiamo di un maggior numero d'informazioni riguardo agli eventi e alla loro gestione. Facciamone un breve riassunto.

MainLoop è una grande porzione di codice che continua a ripetersi, e riceve costantemente da InterruptMain informazioni sull'alternarsi degli eventi. A ogni ciclo, le informazioni ricevute vengono scandite ed elaborate. In particolare vengono individuati gli eventi che riguardano i menu, le icone, la pressione del pulsante del mouse in aree non convenzionali, la pressione dei tasti sulla tastiera e i processi temporizzati.

1) Se l'evento è convenzionale, MainLoop procede alla gestione del menu o dell'icona coinvolta, e in particolare, se si tratta di un evento finale, cede provvisoriamente il controllo alla relativa routine di servizio.

2) Se l'evento non è convenzionale, il controllo viene ceduto alla routine di servizio puntata dal relativo vettore di sistema.

3) Se l'evento è temporizzato, MainLoop controlla lo "stato" del processo, e se è possibile lo manda in esecuzione.

È importante sottolineare che quando una routine di servizio, o un processo, detiene il controllo della CPU, non possono essere elaborati gli altri eventi, dal momento che MainLoop è provvisoriamente inattiva. Le routine di servizio e i processi non dovrebbero quindi rimanere in esecuzione troppo a lungo, e in ogni caso devono tassativamente restituire il controllo a MainLoop. Se una routine di servizio, durante il suo svolgimento, deve attendere che venga effettuata una certa operazione, non fa altro che definire un ulteriore evento e ricedere il controllo a MainLoop.

Da questa sommaria descrizione di MainLoop e dei suoi compiti, possiamo renderci conto quanto sia complessa e flessibile questa routine. Si tratta di un "programma" in grado di adattarsi alle più svariate esigenze. È sufficiente fornire le tavole di definizione di menu, icone e processi temporizzati, aggiornare i vettori per gli eventi non convenzionali che si vogliono elaborare, e lasciare al sistema l'intero controllo del flusso di eventi. Prima dell'avvento di *GEOS* i programmatori del C-64 non avevano mai avuto a disposizione un "cervello" tanto sofisticato e affidabile.

InterruptMain

Almeno una notizia su questa sofisticata routine dovrebbe essere già trapelata: è l'informatrice di MainLoop. Si tratta di una routine di interrupt che viene mandata in esecuzione dal C-64 ogni sessantesimo di secondo, indipendentemente dai codici che la CPU sta eseguendo; il periodo di tempo durante il quale detiene il controllo della CPU è molto minore di 1/60 di secondo, in modo da lasciarle tutto il tempo necessario per l'esecuzione dei normali codici. Il suo compito principale è quello di interagire con l'hardware (in genere la porta per il dispositivo di controllo del mouse, lo sprite del mouse,

tastiera).

MainLoop è una routine molto elaborata, e un suo ciclo completo (che comprende ovviamente l'esecuzione di tutte le routine di servizio corrispondenti agli eventi in corso) può durare anche più di 1/60 di secondo. Sembra quindi che informazioni consecutive, con brevissimi intervalli di tempo l'una dall'altra, possano perdersi. È proprio InterruptMain a evitare questo problema, interagendo con l'hardware del computer ogni sessantesimo di secondo ed evitando così che vadano perdute informazioni come i movimenti del mouse, le pressioni del pulsante o dei tasti della tastiera.

Quando InterruptMain riceve il controllo, salva lo stato della macchina e procede a controllare l'hardware. Vediamo quali sono le principali operazioni a cui si dedica. Se si accorge che l'utente ha premuto un tasto, memorizza il carattere corrispondente in un buffer, cosicché il carattere non venga perso se per caso alla successiva chiamata di interrupt MainLoop non ha avuto ancora il tempo di leggerlo. Parallelamente, imposta un opportuno flag che MainLoop legge a ogni ciclo per sapere se il buffer contiene caratteri in attesa di essere prelevati. Il buffer impiegato è una struttura FIFO (First In First Out) e solo quando si svuota interamente il flag di segnalazione torna allo stato originale per indicare che non ci sono caratteri in attesa di essere elaborati.

InterruptMain procede anche a decrementare i timer dei processi temporizzati che sono diversi da zero, a leggere le informazioni ricevute dal dispositivo di controllo del mouse e a muovere lo sprite corrispondente sullo schermo. La stessa routine, inoltre, prima di aggiornare la posizione del mouse sullo schermo, verifica che la posizione richiesta non sia esterna all'area di movimento consentita al mouse. Quest'area generalmente corrisponde all'intero schermo, ma le applicazioni hanno la facoltà di variarla a piacere per restringere il campo d'azione del mouse. Per esempio, in *geoPublish*, quando si sta disegnando una linea, l'utente non riesce a spostare il mouse all'esterno della pagina riservata al disegno di forme geometriche. Quando InterruptMain si accorge che il mouse cerca di oltrepassare i confini definiti dal sistema o dall'applicazione, imposta un opportuno flag di segnalazione. MainLoop, durante il suo ciclo, controlla ogni volta lo stato di questo flag, e, se è impostato, chiama la routine puntata dal vettore di sistema mouseFaultVec. Per esempio, in *geoWrite* questo vettore è impiegato per effettuare lo scroll orizzontale e verticale. InterruptMain segnala anche le pressioni del pulsante del mouse impostando un flag che MainLoop controlla a ogni ciclo.

Possiamo quindi dire che il primo "organo" di *GEOS* che recepisce un evento è il "sistema nervoso" InterruptMain, che provvede a impostare tutti i flag necessari per descriverlo. Quando termina il suo intervento, ripristina lo stato della macchina precedentemente salvato e ritorna. MainLoop, durante ogni ciclo, accede a tutti i flag di comunicazione e procede a elaborare gli eventi segnalati.

Espansione di MainLoop e di InterruptMain

Anche se MainLoop dovrebbe essere in grado di adattarsi alle più svariate esigenze, può accadere che il programmatore voglia servirsene anche per funzioni completamente diverse. In questo caso si dice che l'applicazione introduce dei "codici non strutturati in eventi".

L'aggiunta di questi codici entra in conflitto con la filosofia del sistema operativo a gestione dei flussi di eventi, e si deve quindi valutare attentamente se con le normali procedure lo scopo può essere ugualmente raggiunto. Per espandere MainLoop, è necessario memorizzare nel vettore di sistema appMain l'indirizzo della routine impiegata per l'espansione. MainLoop la manda in esecuzione durante ogni ciclo, e quindi diventa parte integrante del "cervello". Routine di questo tipo si rendono normalmente necessarie per aggiungere al sistema delle funzioni per il controllo degli eventi, ma la loro realizzazione presenta notevoli difficoltà dal momento che possono intervenire su variabili di sistema che di solito non vengono alterate dalle applicazioni.

Un esempio che chiarisce la necessità di aggiungere un codice a MainLoop è il cambio della forma del mouse quando raggiunge una particolare zona dello schermo. Questa routine, che verrebbe eseguita a ogni ciclo di MainLoop, dovrebbe chiamare la routine di sistema IsMseInRegion per verificare se il mouse si trova all'interno della zona stabilita, nel qual caso dovrebbe provvedere a cambiarne la forma. Sebbene una soluzione di questo genere sia soddisfacente, risente della lenta (in termini di millisecondi!) periodicità di MainLoop e può accadere che spostando molto velocemente il mouse (per esempio con un mouse proporzionale), il cambio della forma avvenga con un certo ritardo rispetto al momento nel quale vengono realmente oltrepassati i limiti imposti. Vedremo che utilizzando invece InterruptMain il problema si può risolvere facilmente.

Come MainLoop, anche InterruptMain può essere abilitata a nuove funzioni dalle applicazioni, grazie all'introduzione di codici non strutturati in eventi, solo che in questo caso la situazione è molto più delicata. Le routine di interrupt vengono eseguite ogni sessantesimo di secondo, e per mantenere il sincronismo fra ogni chiamata, devono completarsi in un tempo molto minore rispetto al periodo di interrupt. Non solo perché i codici normali (codici diversi da quelli di interrupt) non potrebbero più riavere il controllo, ma anche perché se gli interrupt non vengono disabilitati durante l'esecuzione della ro

di interrupt, e questa dura più di 1/60 di secondo, si verificherebbe un'altra chiamata di interrupt mentre il sistema sta già svolgendo una routine di interrupt. La sovrapposizione delle chiamate di interrupt, per quanto non sia teoricamente impossibile, non è prevista dal sistema operativo *GEOS*, e quindi dev'essere evitata. Intervenire espandendo InterruptMain potrebbe produrre un effetto come quello appena descritto, che dev'essere assolutamente evitato. Il programmatore che "aggancia" una routine non strutturata in eventi a InterruptMain dev'essere certo che questo non allunghi troppo il tempo necessario per completare la procedura di interrupt.

Le applicazioni possono espandere InterruptMain sia all'inizio che alla fine. Se si desidera che una routine di interrupt *preceda* l'esecuzione di InterruptMain, è necessario memorizzarne l'indirizzo nel vettore di sistema intTopVector. Se invece la routine di interrupt deve *seguire* InterruptMain, l'indirizzo dev'essere memorizzato nel vettore di sistema intBotVector. Entrambi i vettori, in condizioni normali, contengono un indirizzo nullo.

Per fare un esempio nel quale si dimostra utile l'espansione di InterruptMain, riprendiamo quello analizzato per MainLoop. Se la stessa routine impiegata precedentemente venisse aggiunta a InterruptMain, il controllo sulla posizione del mouse sarebbe molto più frequente (60 volte al secondo) e l'aggiornamento della sua forma avverrebbe sicuramente in tempo reale. Inoltre l'operazione è realizzabile, dal momento che una routine di questo genere sarebbe abbastanza rapida da non compromettere la periodicità degli interrupt.

Un passo alla volta

Pian piano stiamo scoprendo un sistema operativo per un computer a 8 bit che non può che lasciarci stupiti. In questo articolo abbiamo illustrato le funzioni basilari di MainLoop e InterruptMain. In seguito faremo la conoscenza anche delle funzioni secondarie, che in questo articolo introduttivo abbiamo preferito tralasciare per non appesantire l'argomento e non rischiare di creare confusione.

Pensate ai meccanismi con i quali il nostro sistema nervoso richiama l'attenzione del cervello, e come il cervello è capace di ricevere nello stesso istante un numero enorme di informazioni diverse... Per sottolineare ancora questa capacità di *GEOS*, prima di concludere, potremmo dire che il sistema può ricevere una stringa di input dall'utente (con il cursore lampeggiante) mantenendo attiva l'intera struttura dei menu e delle icone, e contemporaneamente visualizzare un orologio sullo schermo mentre un ennesimo processo rende visibile la posizione corrente del mouse. L'utente può digitare parte della stringa di input, poi decidere di esplorare la struttura dei menu, e infine completare la stringa di input. Dal punto di vista dell'utente tutti questi eventi avvengono simultaneamente, e i programmatori possono destreggiarsi con estrema facilità tra le varie alternative. Non male per il "vecchio" C-64. ■

COMUNICATO

La **PIX COMPUTER SERVICE**, con sede in Roma - Via F. D'Ovidio n. 6/c, a seguito di una contestazione sollevata dalla IHT Gruppo Editoriale S.r.l. di Milano

COMUNICA

che i diritti esclusivi di utilizzazione editoriale in tutto il mondo dell'opera "***The AmigaDOS Manual***" – in italiano "***Il manuale dell'AmigaDOS***" – spettano alla IHT Gruppo Editoriale S.r.l.;

COMUNICA

pertanto di aver cessato ogni attività di commercializzazione della versione italiana del manuale dell'AmigaDOS

RICONOSCE

che il diritto di pubblicare e commercializzare in italiano il manuale suddetto spetta in via esclusiva alla IHT Gruppo Editoriale S.r.l.

MARZO 1988
AMIGA

UN ARCHIVIO ELETTRONICO PER IL VOSTRO AMIGA

Mantenere costantemente aggiornata una lista di nominativi, poterla consultare rapidamente e svolgere ricerche parziali è un'esigenza frequente. Questo programma in Amiga BASIC offre un'efficace soluzione al problema

di Marco Costantino

Lo scopo del programma, che occupa in memoria circa 20K ed è scritto totalmente in linguaggio Basic, è di consentire una gestione semplice e accurata degli indirizzi in proprio possesso, integrati da una vasta serie di notizie a cui sono destinate ben 19 linee di schermo, al punto che il programma *Gestione Scambi* può essere paragonato a una gestione clienti evoluta.

Il programma consente la coesistenza di più archivi sullo stesso disco o su più dischi. Ogni archivio può contenere da un minimo di 400 nominativi a un massimo che dipende dalla quantità di memoria disponibile nel sistema. Bisogna però notare che maggiore è il numero di nominativi in memoria, maggiore sarà il tempo impiegato dal sort di stampa, che rimane comunque abbastanza rapido.

Il programma, grazie alla sua elevata flessibilità, consente d'immettere e aggiungere dati al file, variare qualsiasi dato, effettuare ricerche secondo qualsiasi chiave di ricerca anche su più campi contemporaneamente e persino su file esterni residenti su disco, stampare una lista di tutti i dati che soddisfano a un certo pattern (anche su più campi), in un ordine qualunque. Le opzioni di stampa vanno dalla stampa di tutti i nominativi presenti in archivio sotto forma di schede clienti, a quella di indirizzi su buste in modulo continuo (per la gioia dei vari rivenditori che spediscono periodicamente gli aggiornamenti dei loro cataloghi e che di solito devono scrivere a macchina, o peggio a penna, tutti gli indirizzi).

A conclusione di questa breve panoramica sulle caratteristiche del programma, si può facilmente capire che le sue prestazioni sono notevolmente superiori a quelle della solita gestione di una rubrica telefonica: si tratta infatti di un'applicazione particolarmente adatta a una gestione clienti di livello professionale, e ad altre destinazioni analoghe.

Per concludere questa introduzione, spendiamo due parole sul nome che abbiamo scelto: *Gestione Scambi*. Con la parola "scambio" vogliamo intendere un'operazione che dà luogo a entrate o uscite di denaro. Ecco quindi il significato del programma: un prodotto adatto a tutti coloro che hanno bisogno di gestire in modo veloce ed efficace un numero molto elevato di nominativi corredati da altrettante informazioni.

Introduzione al funzionamento del programma

Analizziamo ora la procedura d'installazione del programma. Dopo aver digitato correttamente il listato e averlo salvato su un disco, è necessario impartire la sequenza CTRL-A-A per resettare il computer e assicurarsi che non ci siano altri task in funzione (il programma occupa praticamente tutta la memoria disponibile del computer e se vi fossero in esercizio altri processi verrebbe segnalata la mancanza di memoria).

Per caricarlo è sufficiente selezionarne l'icona due volte. Dopo qualche attimo di attesa per il caricamento vengono visualizzati alcuni messaggi di copyright e il programma entra nel modo operativo.

Cominciamo a lavorare

Dopo aver selezionato l'icona del programma, viene visualizzata una schermata introduttiva con la richiesta di input:

```
Digita il nome dell'archivio da caricare <RETURN> per
default
```

e di seguito il cursore. A questo punto, bisogna inserire il nome di un file, in modo che il programma carichi in memoria il file contenente l'archivio di nominativi. Se l'utente digita il nome di un file inesistente, l'applicazione provvede a crearlo su disco, mentre se preme soltanto il RETURN il computer carica (o crea) il file di default, ossia un file convenzionale chiamato "Archivio".

Supponiamo di scrivere "Prova", e di premere RETURN per terminare l'input. Viene allora visualizzata una scritta che segnala all'utente la presenza dei menu a discesa per interagire con l'applicazione, poi l'indicazione del numero di nominativi correntemente presenti nell'archivio (0 nel caso del file Prova), e infine l'indicazione del numero di byte disponibili.

Uso dei menu a discesa

I menu a discesa presenti all'interno del programma sono equivalenti nel funzionamento a quelli di *Intuition*: compaiono alla pressione del tasto di destra del mouse, e vengono aperti con uno spostamento sulla linea superiore della finestra.

Menu a discesa disponibili all'interno del programma

1) Progetti

Apri
Salva
Salva come
Cancella File su disco
Ricomincia
Esci a Sistema Operativo

2) Inserimento

Inserimento Nominativi
Variazione Nominativi
Annullamento Nominativi
Cancellazione Archivio

3) Ricerca

Ricerca Nominativi
Ricerca Nominativi su Files esterni

4) Stampa

Stampa archivio

5) Trasferimento

Trasferimento nominativi

6) Ordinamento

Ordina Nominativi

Come abbiamo sottolineato, per usare il programma l'utente deve servirsi dei menu a discesa, le cui voci sono riportate nella tavola precedente. La prima operazione da compiere è l'inserimento dei nominativi in archivio. Per far questo si deve selezionare l'opzione "Inserimento Nominativi" presente nel menu Inserimento. Dopo qualche istante di attesa, necessario all'applicazione per inizializzare tutte le variabili che utilizzerà in seguito, compare una scheda suddivisa in diverse sezioni, e un cursore lampeggiante per l'inserimento di un nominativo. Durante il periodo di attesa per l'aggiornamento delle variabili interne, necessario soltanto all'inizio di una sessione di lavoro, non bisogna premere alcun tasto per evitare d'introdurre informazioni indesiderate. Facciamo un esempio che illustra l'impiego di queste schede: alla richiesta del cognome scriviamo Costantino (battiamo RETURN); il cursore si sposta automaticamente nel campo del nome: Marco RETURN; indirizzo: C.so Trento 63 RETURN; città: 38086 Pinzolo (TN) RETURN; Tel.: 0465/51610 RETURN; scheda: CM000001 RETURN (nota: questo spazio è riservato al codice d'identificazione del nominativo, o meglio del cliente). Operati questi inserimenti, il cursore si sposta nell'area per le note, composta da 19 righe. Qui possiamo scrivere qualsiasi cosa, per esempio i vari acquisti del cliente. È importante notare che se si desidera concludere l'inserimento della nota prima che siano state riempite le 19 righe disponibili, è sufficiente battere su di una linea il carattere "*", seguito da RETURN.

Terminato l'inserimento viene visualizzata una richiesta:

```
Vuoi continuare (S/N)?
```

Se a questo punto battiamo il tasto "S" pos...

introdurre un altro nominativo, aumentando a due il numero delle voci in memoria; se invece premiamo "N", viene visualizzata una nuova richiesta:

```
Vuoi effettuare l'ordinamento (S/N)?
```

Se si preme "S" viene effettuato l'ordinamento alfabetico, rispetto al cognome, di tutti i nominativi in memoria, altrimenti si ritorna al menu principale con i messaggi di copyright.

A questo punto viene visualizzata anche l'indicazione del numero di nominativi presenti nel file Prova e la quantità di memoria disponibile.

In seguito possiamo salvare su disco l'archivio di nome Prova aprendo il menu "Progetti", e selezionando l'opzione Salva (non occorrono altri input), oppure l'opzione Salva come (richiede un input). La prima salva l'archivio nel file correntemente aperto, mentre la seconda consente di salvarlo in un altro file dichiarandone il nome. In questo secondo caso viene cambiato anche il nome dell'archivio in memoria.

A questo punto non resta che procedere alla descrizione analitica dei comandi.

Descrizione dei menu a discesa disponibili

Menu Progetti

Apri: questo comando permette di caricare in memoria un file archivio di nominativi presente su disco, aggiungendone i dati a quelli già presenti in memoria. Prima di procedere verrà richiesto se si desidera salvare su disco l'archivio correntemente in memoria.

Salva: il comando permette di riversare su un file di archivio esterno (di nome uguale a quello in memoria) tutti i nominativi in memoria. Attenzione! Il file presente su disco con lo stesso nome verrà irrimediabilmente sovrascritto.

Salva come: questa opzione permette di riversare su un file di archivio esterno tutti i nominativi in memoria. A differenza del comando "Salva", questo richiede il nome del file destinazione per consentire l'archiviazione su disco in un file diverso da quello d'origine. Questa opzione cambia anche il nome dell'archivio in memoria. Attenzione! Se è presente su disco un file con lo stesso nome, verrà irrimediabilmente sovrascritto.

Cancella File su disco: questa opzione consente di cancellare qualsiasi file presente su disco. Viene richiesto in input il nome del file da cancellare. Attenzione! Non viene richiesta nessuna conferma e il file cancellato diventa irrecuperabile.

Ricomincia: questa opzione consente di inizializzare nuovamente l'applicazione, e quindi ricominciare l'esecuzione del programma, con la conseguente perdita di tutti i nominativi in memoria. In input viene chiesto se si desidera salvare su disco l'archivio correntemente in memoria. Attenzione! Non viene richiesta alcuna conferma per procedere a inizializzare nuovamente l'applicazione. Archivi di nominativi non salvati su disco vengono cancellati dalla memoria e quindi irrimediabilmente persi.

Esci a Sistema Operativo: questa opzione permette di uscire dal programma restituendo il controllo al *Workbench*. In input viene richiesto se si desidera salvare su disco l'archivio dei nominativi correntemente in memoria. Attenzione! Non viene richiesta nessuna conferma e l'archivio in memoria viene integralmente cancellato.

Menu Inserimento

Inserimento Nominativi: questa opzione permette l'inserimento di più nominativi (record) nell'archivio in memoria. Viene visualizzato un cursore lampeggiante sul campo selezionato (cognome, nome...). Per procedere è necessario premere RETURN. Nelle 19 righe per la nota è possibile terminare l'input battendo su una riga il carattere "*". Al termine di ogni inserimento viene visualizzata la richiesta:

```
Vuoi Continuare (S/N)?
```

Rispondendo "S" si procede a inserire un altro nominativo, mentre rispondendo "N" viene visualizzata la richiesta:

```
Vuoi effettuare l'ordinamento (S/N)?
```

Rispondendo "S" viene effettuato l'ordinamento alfabetico (rispetto al cognome) dei nominativi in memoria e si esce dall'opzione d'inserimento per tornare al menu principale; in alternativa, premendo "N" si torna al menu principale senza che l'archivio venga ordinato.

Nota: i tasti speciali utilizzabili durante l'inserimento dei dati sono:

Backspace (cancella un carattere a sinistra)
Del (cancella un carattere a destra)
F1 (porta all'inizio della linea)
F10 (porta alla fine della linea)
ESC (cancella la linea)
Tasti cursore orizzontali (movimento orizzontale sulla linea)

Nota: durante l'inserimento, l'unico campo che deve obbligatoriamente essere aggiornato è

cognome, mentre i campi successivi possono essere lasciati privi di caratteri.

Variazione Nominativi: questa opzione consente la modifica di ogni nominativo presente in memoria. Prima della modifica viene individuato il nominativo richiesto tramite la ricerca selettiva (vedere a questo proposito la descrizione dell'opzione Ricerca Nominativi). Quando l'applicazione ha trovato il nominativo richiesto, viene visualizzata la sua scheda. Per procedere alle modifiche, è necessario utilizzare le stesse regole descritte per l'opzione Inserimento Nominativi. Attenzione! È necessario evitare di annullare il cognome, pena errori in successive fasi di ordinamento.

Annullamento Nominativi: questa opzione consente la cancellazione di singoli nominativi. L'identificazione del nominativo avviene tramite la ricerca selettiva (vedere a tal proposito la descrizione dell'opzione Ricerca Nominativi). Nota: viene richiesta conferma dell'operazione. Attenzione! Nominativi cancellati non potranno essere recuperati in seguito.

Cancellazione Archivio: questa opzione permette la cancellazione dell'intero archivio di nominativi presente in memoria. Nota: viene richiesta la conferma dell'operazione. Attenzione! Gli archivi cancellati non possono essere recuperati in seguito.

Menu Ricerca

Ricerca Nominativi: questa opzione permette di effettuare la ricerca dei nominativi tramite diverse chiavi di ricerca. Dopo aver selezionato l'opzione vengono visualizzati tutti i possibili modi di ricerca. Per scegliere l'opzione desiderata è sufficiente selezionarla con il mouse.

– Digitando un nome specifico: questa opzione attiva il modo di ricerca selettiva che viene impiegato anche nella maggior parte delle altre funzioni del programma. Viene visualizzata la scheda di input e alla richiesta:

```
Cosa Conosci?
```

visibile in verde nella parte bassa dello schermo, si deve inserire nell'opportuno campo la parte di nominativo di cui si è a conoscenza. Facciamo un esempio: se di un nominativo si conosce soltanto una parte del numero telefonico, ci si deve portare sul campo "Tel." (tramite successioni di RETURN) e battere il numero telefonico, o almeno la parte di cui si è a conoscenza. A questo punto il computer visualizza tutti i nominativi aventi il numero telefonico, o parte di esso, equivalente a quello digitato. Nel caso che nessun nominativo rispondesse a queste caratteristiche, verrà visualizzato il messaggio:

```
Nominativo non Trovato!!
```

– Digitando una lettera dell'alfabeto: questa opzione permette la ricerca di tutti i nominativi in memoria aventi il cognome che inizia con la lettera battuta.

– Visualizza una lista dei nominativi: tramite questa opzione, viene visualizzata una lista dei nominativi in memoria (in altissima risoluzione, 640*512), ridotta ai soli campi Codice (o scheda), cognome, nome, città. Nota: i tasti utilizzabili sono ESC per uscire, e un tasto qualsiasi per procedere nella visualizzazione delle pagine successive.

– Visualizza tutti i nominativi: vengono visualizzati in successione tutti i nominativi presenti in memoria.

– Ritorna al menu principale: si esce dall'opzione ricerca ritornando al menu principale.

Ricerca di nominativi su file esterni: questa opzione permette di cercare e caricare in memoria nominativi presenti in archivi (o meglio in file di archivio) residenti su disco. Il sistema di ricerca è quello selettivo, e i tempi sono molto ridotti. L'opzione richiede in input i dati per svolgere la ricerca selettiva, e i nomi degli archivi (massimo 10) su cui effettuare la ricerca. Nota: è possibile indicare archivi inesistenti, dal momento che il programma provvede a crearli.

Menu Stampa

Stampa dell'archivio: questa opzione permette di stampare tutti i nominativi, o una parte tramite la ricerca selettiva, in diversi modi.

– Tutti i nominativi con dati relativi: consente di stampare integralmente tutte le schede dei nominativi presenti in memoria.

– Tutti i nominativi: vengono stampate tutte le schede dei nominativi presenti in memoria senza le 19 linee di nota, utilizzando quindi solo due linee per nominativo.

– Selezione di nominativi con dati relativi: come per l'opzione "Tutti i nominativi con dati relativi", ma attraverso la ricerca selettiva.

– Selezione di nominativi: come per l'opzione "Tutti i nominativi", ma attraverso la ricerca selettiva.

– **Tutti gli indirizzi:** permette di stampare tutti i nominativi in memoria in forma di indirizzi postali, garantendo così la possibilità di utilizzare buste in modulo continuo. In input viene chiesto il margine sinistro.

– **Selezione di indirizzi:** come per l'opzione precedente, ma attraverso la ricerca selettiva.

– **Ritorno al Menu principale:** si esce dall'opzione di stampa ritornando al menu principale.

Menu Trasferimento

Trasferimento nominativi: questa opzione permette il trasferimento di tutti i nominativi in memoria, o di una parte, tramite la ricerca selettiva. In pratica si aggiungono i record prescelti a un file esterno su disco. Sono disponibili due modi di trasferimento:
1) tutti i nominativi verso un file
2) selezione di nominativi verso un file (attraverso la ricerca selettiva).

In entrambi i casi viene richiesto in input il file destinazione, al quale verranno aggiunti i record.

Menu Ordinamento

Ordina i nominativi: questa opzione permette il riordinamento alfabetico (rispetto al cognome) di tutti i nominativi in memoria. L'algoritmo di ordinamento è abbastanza rapido, ma per archivi molto disordinati e di grandi dimensioni si rende ugualmente necessaria una piccola attesa. Durante questo periodo viene visualizzato un messaggio di attesa e alcune indicazioni inerenti all'ordinamento in corso.

IMPORTANTE! Non si deve fermare il programma per nessuna ragione, dato che si perderebbero inevitabilmente i dati contenuti nell'archivio in memoria. Nell'eventualità che il programma si blocchi, digitare "gosub salva", in modo da risparmiare i dati presenti in memoria. In ogni caso, per terminare la sessione di lavoro e uscire dal programma si utilizzi esclusivamente l'opzione nel menu Progetti, "Esci a Sistema Operativo".

Personalizzazione del programma

Per utilizzare il programma è sufficiente digitare il listato modificando le seguenti linee in modo da ricostruirlo a seconda delle diverse esigenze e configurazioni di memoria.

```
-linea1: CLEAR ,130000&
```

se si possiede un Amiga 2000 sostituire con:

```
CLEAR ,320000&
```

```
-label DIMVARIABILI, prima linea: massimo%=400
```

se si possiede un Amiga 2000 sostituire con:

```
massimo%=1000
```

```
-SUB InputC, label accettacarattere, linea11: IF codice%>
96 and codice%<123 THEN codice%=(codice%-32)
```

eliminare questa linea per ottenere anche l'input delle minuscole.

Note tecniche

Il programma *Gestione Scambi* è scritto totalmente in Basic e occupa circa 20K. È stato realizzato in maniera molto strutturata (fa uso intenso di subroutine e SUB) e per questo si presta facilmente a essere ulteriormente modificato dall'utente che desidera personalizzarlo. Le variabili principali e le matrici del programma sono:

Arch$	= nome dell'archivio corrente
I	= numero di nominativi attualmente in memoria
U	= cicli FOR-NEXT e cicli di ritardo
A$	= cicli INKEY$ e I/O data
Z	= variabile utilizzata come puntatore della matrice NT$(massimo%,Z)
massimo%	= dimensionamento dinamico della matrice NT$ (massimo%,Z)
menuid	= funzione MENU(0)
menuit	= funzione MENU(1)

matrici:

NT$(massimo%,Z)	= contiene tutto l'archivio in memoria
N$(25)	= utilizzata come matrice temporanea nella ricerca su disco

Listato: *Gestione Scambi*

```
' Gestione Scambi
' Written by Marco Costantino
' Thanks to Giacomo Maestranzi and Albert
o Turrini for Graphics
' Pubblicato da Commodore Gazette
'
' Per utilizzare il programma battere il
presente listato,
' salvarlo e clickare la sua Icona da Wor
kbench

CLEAR ,130000& 'modificare a seconda dell
e configurazioni di memoria

inizio:
IF massimo%=0 THEN GOSUB DIMVARIABILI
GOSUB START
GOSUB INITMENU
GOSUB SCEGLIARCH
GOSUB LOADARCH
GOTO ABOUT

DIMVARIABILI:
massimo%=400 'modificare a seconda delle
configurazioni di memoria
DIM nt$(massimo%,25),n$(25)
RETURN

START:
SCREEN 1,640,256,2,2
WINDOW 1,"Gestione Scambi - Written by Ma
rco Costantino - Commodore Gazette - Rev
3.41",(0,1)-(6
30,241),0,1
PALETTE 0,0,0,0: PALETTE 1,1,0,0
PALETTE 2,1,1,0: PALETTE 3,0,1,0
RETURN

INITMENU:
MENU 1,0,1,"Progetti"
MENU 1,1,1,"Apri                 "
MENU 1,2,1,"Salva                "
MENU 1,3,1,"Salva come           "
MENU 1,4,1,"Cancella File su Disco "
MENU 1,5,1,"Ricomincia"
MENU 1,6,1,"Esci a Sistema Operativo"
MENU 2,0,1,"Inserimento"
MENU 2,1,1,"Inserimento  Nominativi"
MENU 2,2,1,"Variazione   Nominativi"
MENU 2,3,1,"Annullamento Nominativi"
MENU 2,4,1,"Cancellazione Archivio "
MENU 3,0,1,"Ricerca"
MENU 3,1,1,"Ricerca Nominativi
 "
MENU 3,2,1,"Ricerca Nominativi su Files e
sterni"
MENU 4,0,1,"Stampa"
MENU 4,1,1,"Stampa archivio"
MENU 5,0,1,"Trasferimento"
MENU 5,1,1,"Trasferimento nominativi"
MENU 6,0,1,"Ordinamento"
MENU 6,1,1,"Ordina Nom."
ON MENU GOSUB FINDMENU
RETURN

SCEGLIARCH:
CALL MASCHERA
LOCATE 23,1:COLOR 3
INPUT "Digita il nome dell'archivio da ca
ricare <RETURN> per default ",Arch$
IF Arch$="" THEN Arch$="Archivio"
RETURN

LOADARCH:
IF FRE(0)<=6000 OR I=>massimo%-10 THEN CL
OSE#1:CLS:PRINT "Memoria insufficiente!":
FOR u=0 TO 100
00:NEXT:RETURN
OPEN Arch$ FOR APPEND AS #1:CLOSE#1
OPEN Arch$ FOR INPUT AS #1
WHILE NOT EOF(1)
  I=I+1:LOCATE 1,1:PRINT I
  IF I/10=INT(I/10) THEN IF FRE(0)<=6000
OR I>=massimo%-10 THEN CLOSE#1:I=I-1:CLS:
PRINT "Archivi
o su disco troppo grande!":FOR u=0 TO 100
00:NEXT:RETURN
  FOR Z=1 TO 25:INPUT#1,nt$(I,Z)
  NEXT
WEND
CLOSE#1:RETURN

ABOUT:
MENU OFF
CALL MASCHERA
MENU ON
963 LOCATE 23,16:COLOR 3
PRINT "Utilizzare i menu a discesa per co
ntinuare."
FOR u=0 TO 1000:NEXT
LOCATE 23,16:PRINT SPACE$(43);
FOR u=0 TO 1000:NEXT
GOTO 963

FINDMENU:
Menuid=MENU(0):menuit=MENU(1)
ON Menuid GOTO PROJECT,INSERT,RICERCO,STA
MPO,TRASFERISCI,SORT

PROJECT:
ON menuit GOTO Caricarch,Savearch,savearc
hAS,Killer,Restart,Quit

Caricarch:
CLS:LOCATE 23,18:COLOR 3:PRINT "Vuoi salv
are l'archivio su disco (S/N) ?"
1 a$=INKEY$
IF a$="S" OR a$="s" THEN GOSUB SALVA: GOT
O continua
IF a$="N" OR a$="n" THEN GOTO continua
GOTO 1
continua:
GOSUB SCEGLIARCH:GOSUB LOADARCH:CALL MASC
HERA:RETURN

Savearch:
GOSUB SALVA
RETURN

Restart:
CLS:LOCATE 23,18:COLOR 3:PRINT  "Vuoi sal
vare l'archivio su disco (S/N) ?"
2 a$=INKEY$:IF a$="" THEN 2
IF a$="S" OR a$="s" THEN GOSUB SALVA
FOR I1=1 TO I
  FOR Z=1 TO 25
    nt$(I1,Z)=""
  NEXT
NEXT
I=0:Arch$="":GOTO inizio

savearchAS:
CLS:COLOR 3:INPUT "Qual e' il nome con cu
i vuoi salvare il corrente archivio ";Arc
h$:IF Arch$=""
 THEN CALL MASCHERA:RETURN
GOSUB SALVA
CALL MASCHERA:RETURN

Killer:
file$="":CLS:PRINT:COLOR 3:INPUT "Digita
 il nome del file da cancellare su disco "
;file$
IF file$="" THEN CALL MASCHERA:RETURN
OPEN file$ FOR APPEND AS #1:CLOSE #1
KILL file$
CALL MASCHERA:RETURN

Quit:
CLS:LOCATE 23,18:PRINT "Vuoi salvare l'ar
chivio su disco (S/N) ?"
```

```
3 a$=INKEY$
IF a$="S" OR a$="s" THEN GOSUB SALVA:SYST
EM
IF a$="N" OR a$="n" THEN SYSTEM
GOTO 3

INSERT:

ON menuit GOTO Insertno,Variazno,Annullno
,Canarch

Insertno:
CALL MASCHERA2
I=I+1
IF FRE(0)<=6000 OR I>=massimo%-10 THEN I=
I-1:CLS:PRINT "Memoria esaurita!!! Devi i
nizializzare u
n nuovo archivio!!!":FOR u=0 TO 10000:NEX
T:CALL MASCHERA:RETURN
imm$="":o=14:v=2:b%=30:GOSUB chiamaI:nt$(
I,1)=imm$:imm$="":IF nt$(I,1)="" THEN I=I
-1: GOTO Inser
tno
o=53:v=2:b%=21: GOSUB chiamaI:nt$(I,2)=im
m$:imm$=""
o=14:v=4:b%=60: GOSUB chiamaI:nt$(I,3)=im
m$:imm$=""
o=14:v=6:b%=30: GOSUB chiamaI:nt$(I,4)=im
m$:imm$=""
o=53:v=6:b%=21: GOSUB chiamaI:nt$(I,5)=im
m$:imm$=""
o=39:v=8:b%=8 : GOSUB chiamaI:nt$(I,6)=im
m$:imm$=""
FOR Z=7 TO 25
  o=2:v=9+Z-6:b%=73:GOSUB chiamaI: IF imm
$="*" THEN imm$="": GOTO fine ELSE nt$(I,
Z)=imm$:imm$="
"
NEXT
fine:
LOCATE 25,27:COLOR 3:PRINT "Vuoi continua
re (S/N)?"
4 a$=INKEY$
IF a$="S" OR a$="s" THEN GOTO Insertno
IF a$="N" OR a$="n" THEN sceglio rd
GOTO 4
sceglio rd:
LOCATE 25,19:PRINT SPACE$(28):LOCATE 25,2
0:COLOR 3:PRINT "Vuoi effettuare l'ordina
mento (S/N)?"
5 a$=INKEY$
IF a$="S" OR a$="s" THEN GOSUB ORDINAMENT
O:CALL MASCHERA:RETURN
IF a$="N" OR a$="n" THEN CALL MASCHERA:RE
TURN
 GOTO 5

Variazno:
flag%=0:CLS:COLOR 3:IF I=0 THEN LOCATE 25
,30:PRINT "L'archivio e' vuoto!!":FOR u=0
 TO 10000:NEXT
:CALL MASCHERA:RETURN
GOSUB nome:IF flag%=1 THEN CALL MASCHERA:
RETURN
CALL MASCHERA2
imm$=""
o=13:v=2:b%=30:imm$=nt$(I1,1):GOSUB chiam
aI:nt$(I1,1)=imm$
o=52:v=2:b%=21:imm$=nt$(I1,2):GOSUB chiam
aI:nt$(I1,2)=imm$
o=13:v=4:b%=60:imm$=nt$(I1,3):GOSUB chiam
aI:nt$(I1,3)=imm$
o=13:v=6:b%=30:imm$=nt$(I1,4):GOSUB chiam
aI:nt$(I1,4)=imm$
o=52:v=6:b%=21:imm$=nt$(I1,5):GOSUB chiam
aI:nt$(I1,5)=imm$
o=38:v=8:b%=8 :imm$=nt$(I1,6):GOSUB chiam
aI:nt$(I1,6)=imm$
FOR Z=7 TO 25
  o=2:v=9+Z-6:b%=73:imm$=nt$(I1,Z):GOSUB
chiamaI:IF imm$="*" THEN CALL MASCHERA:RE
TURN ELSE nt$(
I1,Z)=imm$
NEXT
```

```
CALL MASCHERA:RETURN

Annullno:
flag%=0:i2=0:I1=0:CLS:COLOR 3:IF I=0 THEN
 LOCATE 25,30:PRINT "L'archivio e' vuoto!
!":FOR u=0 TO
10000:NEXT:CALL MASCHERA:RETURN
GOSUB nome:IF flag%=1 THEN CALL MASCHERA:
RETURN
COLOR 3:LOCATE 25,22:PRINT SPACE$(46):LOC
ATE 25,31: PRINT "Sei sicuro (S/N)?"
6 a$=INKEY$
IF a$="S" OR a$="s" THEN nt$(I1,1)=CHR$(2
7):GOSUB ORDINAMENTO:CALL MASCHERA:RETURN
IF a$="N" OR a$="n" THEN CALL MASCHERA:RE
TURN
GOTO 6

Canarch:
flag%=0:CLS:COLOR 3:IF I=0 THEN LOCATE 25
,30:PRINT "L'archivio e' vuoto!!":FOR u=0
 TO 10000:NEXT
:CALL MASCHERA:RETURN
COLOR 3:LOCATE 25,31:PRINT "Sei sicuro (S
/N) ?"
7 a$=INKEY$
IF a$="S" OR a$="s" THEN
  FOR I1=1 TO I
    FOR Z=1 TO 25
      nt$(I1,Z)=""
    NEXT
  NEXT
  I=0:CALL MASCHERA:RETURN
END IF
IF a$="N" OR a$="n" THEN CALL MASCHERA:RE
TURN
GOTO 7

RICERCO:
ON menuit GOTO ricerca,nome1
ricerca:
CLS:COLOR 3:IF I=0 THEN LOCATE 25,30:PRIN
T "L'archivio e' vuoto!!":FOR u=0 TO 100
00:NEXT:CALL M
ASCHERA:RETURN
n$(1)="1. Digitando un nome specifico"
n$(2)="2. Digitando una lettera dell'alfa
beto"
n$(3)="3. Visualizza la lista dei nominat
ivi
n$(4)="4. Visualizza tutti i nominativi"
n$(5)="5. Ritorna al menu principale"
n=5:stringa$="          Ricerca nominativi:
":GOSUB MASCHERA3
IF p=1 THEN GOSUB nome:GOTO ricerca
IF p=2 THEN GOSUB Lettera:GOTO ricerca
IF p=3 THEN GOSUB start1:GOSUB lista:GOSU
B START:GOTO ricerca
IF p=4 THEN GOSUB tutto:GOTO ricerca
IF p=5 THEN CALL MASCHERA:RETURN
lista:
i2=1:i3=54:flag%=0
364 IF i2>I THEN GOSUB tasti:RETURN
IF i2<>1 THEN GOSUB tasti:IF flag%=1 THEN
 RETURN
CALL maschera4
FOR I1=i2 TO i3:IF (I1-i2)>=I THEN GOSUB
tasti:RETURN
nt$(0,6)=MID$(nt$(I1,6),1,6)
COLOR 2,0
LOCATE I1+5-(i3-54),3:PRINT I1
LOCATE I1+5-(i3-54),10:PRINT nt$(0,6)
LOCATE I1+5-(i3-54),19:PRINT nt$(I1,1)
LOCATE I1+5-(i3-54),37:PRINT nt$(I1,2)
LOCATE I1+5-(i3-54),55:PRINT nt$(I1,4)
GOSUB tasta:IF flag%=1 THEN RETURN
NEXT:i2=I1:i3=i3+54:GOTO 364
RETURN
tasta:
flag%=0
a$=INKEY$:IF a$="" THEN RETURN
IF ASC(a$)=27 THEN flag%=1
```

```
RETURN
tasti:
flag%=0
365 a$=INKEY$:IF a$="" THEN 365
IF ASC(a$)=27 THEN flag%=1:RETURN
RETURN
ritorno:
LOCATE 25,20:COLOR 3:PRINT  SPACE$(50):LO
CATE 25,27:PRINT  "Nominativo non trovato
!!":FOR u=0 TO
 10000:NEXT:flag%=1:RETURN
nome:
IF I=0 THEN LOCATE 25,30:PRINT  "L'archiv
io e' vuoto!!":FOR u=0 TO 10000:NEXT:CALL
 MASCHERA:RETU
RN
IF flag%=1 THEN GOTO Ciclo
i2=0:I1=i2:imm$=""
CALL MASCHERA2
LOCATE 25,33:COLOR 3:PRINT  "Cosa conosci
?"
o=14:v=2:b%=30:GOSUB chiamaI:nt$(0,1)=imm
$:imm$=""
o=53:v=2:b%=21:GOSUB chiamaI:nt$(0,2)=imm
$:imm$=""
o=14:v=4:b%=60:GOSUB chiamaI:nt$(0,3)=imm
$:imm$=""
o=14:v=6:b%=30:GOSUB chiamaI:nt$(0,4)=imm
$:imm$=""
o=53:v=6:b%=21:GOSUB chiamaI:nt$(0,5)=imm
$:imm$=""
o=39:v=8:b%=8 :GOSUB chiamaI:nt$(0,6)=imm
$:imm$=""
FOR Z=7 TO 25
  imm$="":o=2:v=9+Z-6:b%=73:GOSUB chiamaI
 :IF imm$="*" THEN Ciclo ELSE nt$(0,Z)=imm
$
NEXT
Ciclo:
I1=I1+1:IF I1>I THEN ritorno
FOR Z=1 TO 25
  IF nt$(0,Z)<>"" THEN IF MID$(nt$(I1,Z),
1,LEN(nt$(0,Z)))=nt$(0,Z) THEN GOSUB vis:
GOTO 9 ELSE Ci
clo
NEXT
GOTO Ciclo
9 a$=INKEY$
IF a$="S" OR a$="s" THEN RETURN
IF a$="N" OR a$="n" THEN Ciclo
GOTO 9
Lettera:
I1=0:CLS:PRINT :COLOR 3:INPUT "Qual e' la
 lettera dell'alfabeto ";le$:IF le$="" TH
EN Lettera ELS
E IF ASC(le$)<=122 AND ASC(le$)=>97 THEN
Legrande$=CHR$(ASC(le$)-32) ELSE Legrande
$=CHR$(ASC(le$
)+32)
Ciclo2:
I1=I1+1: IF I1>I THEN ritorno
  IF MID$(nt$(I1,1),1,1)=le$ OR MID$(nt$(
I1,1),1,1)=Legrande$ THEN GOSUB vis:GOTO
10
  GOTO Ciclo2
 10 a$=INKEY$
IF a$="S" OR a$="s" THEN RETURN
IF a$="N" OR a$="n" THEN GOTO Ciclo2
GOTO 10
tutto:
I1=0
Ciclo3:
I1=I1+1:IF I1>I THEN ritorno
GOSUB vis: GOTO 11
GOTO Ciclo3
11 a$=INKEY$
IF a$="S" OR a$="s" THEN RETURN
IF a$="N" OR a$="n" THEN GOTO Ciclo3
GOTO 11
selezarchivi:
CLS:COLOR 3,0:LOCATE 2,10:PRINT "Scrivi i
 nomi degli archivi sui quali effettuare
 la ricerca":LO
CATE 3,28:PRINT "<RETURN> per terminare"
PRINT :FOR I1=1 TO 10
  COLOR 3,0:PRINT "Archivio "I1": ";:COLO
R 2,0:INPUT " ",Arch$(I1):IF Arch$(I1)=""
 THEN n=I1-1:R
ETURN
NEXT:IF I1>=10 THEN n=10 ELSE n=I1:RETURN
nome1:
flug%=1:CALL MASCHERA2
LOCATE 25,33:COLOR 3:PRINT  "Cosa conosci
?"
o=14:v=2:b%=30:GOSUB chiamaI:nt$(0,1)=imm
$:imm$=""
o=53:v=2:b%=21:GOSUB chiamaI:nt$(0,2)=imm
$:imm$=""
o=14:v=4:b%=60:GOSUB chiamaI:nt$(0,3)=imm
$:imm$=""
o=14:v=6:b%=30:GOSUB chiamaI:nt$(0,4)=imm
$:imm$=""
o=53:v=6:b%=21:GOSUB chiamaI:nt$(0,5)=imm
$:imm$=""
o=39:v=8:b%=8 :GOSUB chiamaI:nt$(0,6)=imm
$:imm$=""
FOR Z=7 TO 25
  imm$="":o=2:v=9+Z-6:b%=73:GOSUB chiamaI
 :IF imm$="*" THEN imm$="":GOTO 386 ELSE n
t$(0,Z)=imm$
NEXT
386 GOSUB selezarchivi
FOR I1=1 TO n
CLS:LOCATE 25,34:COLOR 3:PRINT "Attendere
!!!":LOCATE 1,1:PRINT "Sto ricercando nel
 file '";Arch$
(I1)"'"
CLOSE#1:OPEN Arch$(I1) FOR APPEND AS #1:C
LOSE #1
  OPEN Arch$(I1) FOR INPUT AS #1
  WHILE NOT EOF(1)
    FOR Z=1 TO 25
      INPUT#1,n$(Z)
      IF nt$(0,Z)<>"" THEN IF MID$(n$(Z),
1,LEN(nt$(0,Z)))=nt$(0,Z) THEN flog%=1
    NEXT
    IF flog%=1 THEN
      flog%=0:GOSUB vis
      pro: a$=INKEY$:IF a$="" THEN pro
      IF a$="S" OR a$="s" THEN IF FRE(0)>
6000 AND I<massimo%-10 THEN I=I+1:FOR Z=1
 TO 25:nt$(I,Z
)=n$(Z):NEXT:CLOSE#1:flug%=0:CALL MASCHER
A:RETURN ELSE CALL MASCHERA:RETURN
      flog%=0
    END IF
  WEND
NEXT
LOCATE 25,10:COLOR 3:PRINT SPACE$(70):LOC
ATE 25,27:PRINT "Nominativo non trovato!!
":FOR u=0 TO 1
0000:NEXT:CALL MASCHERA:RETURN
CLOSE#1:flug%=0:CALL MASCHERA:RETURN
STAMPO:
CLS:IF I=0 THEN COLOR 3:LOCATE 25,30:PRIN
T  "L'archivio e' vuoto!!":FOR u=0 TO 100
00:NEXT:CALL M
ASCHERA:RETURN
COLOR 3
n$(1)="1. Tutti i nominativi con dati rel
ativi"
n$(2)="2. Tutti i nominativi"
n$(3)="3. Selezione di nom. con dati rela
tivi"
n$(4)="4. Selezione di nominativi"
n$(5)="5. Tutti gli indirizzi"
n$(6)="6. Selezione di indirizzi"
n$(7)="7. Ritorna al menu principale"
n=7:stringa$="          Stampa:":GOSUB MAS
CHERA3
IF p=1 THEN GOSUB tuttire:GOTO STAMPO
IF p=2 THEN GOSUB tuttise: GOTO STAMPO
IF p=3 THEN GOSUB unore   : GOTO STAMPO
IF p=4 THEN GOSUB unose   : GOTO STAMPO
IF p=5 THEN GOSUB indirizzi:GOTO STAMPO
IF p=6 THEN GOSUB indirizzo:GOTO STAMPO
```

```
IF p=7 THEN CALL MASCHERA:RETURN

tuttire:
CLS:PRINT:COLOR 3:PRINT "Premi un tasto p
er stampare"
WHILE INKEY$=""
WEND
CLS:PRINT "Printing...attendere"
OPEN "PRT:" FOR OUTPUT AS #1
FOR I1=1 TO I
  PRINT #1,nt$(I1,1);:PRINT #1," ";:PRINT
 #1,nt$(I1,2)
  PRINT #1,nt$(I1,3)
  PRINT #1,nt$(I1,4)
  PRINT #1,"tel.  ";:PRINT #1,nt$(I1,5)
  PRINT #1,""
  PRINT #1,""
  PRINT #1,SPACE$(27);"Scheda ";nt$(I1,6)
  FOR u=1 TO 69:PRINT #1,"-";:NEXT:PRINT
#1,"-"
  FOR Z=7 TO 25:PRINT #1,nt$(I1,Z):NEXT
  FOR u=1 TO 69:PRINT #1,"-";:NEXT:PRINT
#1,"-"
  PRINT #1,"":PRINT #1,"":PRINT #1,"":PRI
NT #1,"":PRINT #1,""
NEXT:CLOSE#1:RETURN

tuttise:
CLS:PRINT:COLOR 3:PRINT  "Premi un tasto
per stampare"
WHILE INKEY$=""
WEND
CLS:PRINT  "Printing...attendere"
OPEN "PRT:" FOR OUTPUT AS #1
FOR I1=1 TO I
  PRINT #1,nt$(I1,6);"      ";nt$(I1,1);"
 ";nt$(I1,2)
  PRINT #1,nt$(I1,3);"  ";nt$(I1,4);"  ";
nt$(I1,5)
  FOR u=1 TO 69:PRINT #1,"-";:NEXT:PRINT
#1,"-"
NEXT
CLOSE#1:RETURN
unore:
fleg%=0:i2=0:I1=0
loopfl:
i2=I1:flag%=0:GOSUB nome:IF flag%=1 THEN
i2=0:RETURN
IF i2=0 THEN CLS:PRINT :COLOR 3:PRINT  "P
remi un tasto per stampare" ELSE printing
1
WHILE INKEY$=""
WEND
printing:
CLS:PRINT "Printing...attendere"
OPEN "PRT:" FOR OUTPUT AS #1
PRINT #1,nt$(I1,1);:PRINT #1," ";:PRINT #
1,nt$(I1,2)
PRINT #1,nt$(I1,3)
PRINT #1,nt$(I1,4)
PRINT #1,"Tel. ";:PRINT #1,nt$(I1,5)
PRINT #1,"":PRINT #1,""
PRINT #1,SPACE$(27);"Scheda ";nt$(I1,6)
FOR u=1 TO 69:PRINT #1,"-";:NEXT:PRINT #1
,"-"
FOR Z=7 TO 25:PRINT #1,nt$(I1,Z):NEXT
FOR u=1 TO 69:PRINT #1,"-";:NEXT:PRINT #1
,"-"
PRINT #1,"":PRINT #1,"":PRINT #1,"":PRINT
 #1,"":PRINT #1,""
CLOSE#1:fleg%=1:GOTO loopfl
unose:
fleg%=0:i2=0:I1=0
loopfl2:
I1=I1:flag%=0:GOSUB nome:IF flag%<>0 THEN
 i2=0:RETURN
IF i2=0 THEN CLS:PRINT :COLOR 3:PRINT "Pr
emi un tasto per stampare" ELSE printing2
WHILE INKEY$=""
WEND
printing2:
CLS:PRINT "Printing...attendere"
OPEN "PRT:" FOR OUTPUT AS #1
PRINT #1,nt$(I1,6);"    ";nt$(I1,1);" "
;nt$(I1,2)
PRINT #1,nt$(I1,3);"  ";nt$(I1,4);"  ";nt
$(I1,5)
CLOSE#1:fleg%=1:GOTO loopfl2
indirizzi:
CLS:a=0:COLOR 3:PRINT :INPUT "Quante colo
nne vuoi saltare ";a
FOR I1=1 TO I
OPEN "PRT:" FOR OUTPUT AS #1
  GOSUB tasto
  CLS:PRINT "Printing...attendere"
  PRINT#1,SPACE$(a);"Egregio Signor    ";
nt$(I1,6)
  PRINT #1,""
  PRINT #1,SPACE$(a);nt$(I1,1);" ";nt$(I1
,2)
  PRINT #1,SPACE$(a);nt$(I1,3)
  PRINT #1,SPACE$(a);nt$(I1,4)
CLOSE#1:NEXT:RETURN
tasto:
IF i2=0 THEN CLS:PRINT :PRINT "Posiziona
la carta e premi un tasto per stampare" E
LSE RETURN
WHILE INKEY$=""
WEND
RETURN
indirizzo:
fleg%=0:I1=0:i2=0:a=0:CLS:PRINT :COLOR 3:
INPUT "Quante colonne vuoi saltare ";a
loopfl3:
i2=I1:flag%=0:GOSUB nome:IF flag%<>0 THEN
 i2=0:RETURN
GOSUB tasto:CLS:COLOR 3:PRINT "Printing..
.attendere"
OPEN "PRT:" FOR OUTPUT AS #1
PRINT #1,SPACE$(a);"Egregio Signor    ";
nt$(I1,6)
PRINT #1,""
PRINT #1,SPACE$(a);nt$(I1,1);"  ";nt$(I1,
2)
PRINT #1,SPACE$(a);nt$(I1,3)
PRINT #1,SPACE$(a);nt$(I1,4)
CLOSE#1:fleg%=1:GOTO loopfl3
vis:
CALL MASCHERA2
COLOR 2
LOCATE 2,14:IF flug%=1 THEN PRINT n$(1) E
LSE PRINT nt$(I1,1)
LOCATE 2,53:IF flug%=1 THEN PRINT n$(2) E
LSE PRINT nt$(I1,2)
LOCATE 4,14:IF flug%=1 THEN PRINT n$(3) E
LSE PRINT nt$(I1,3)
LOCATE 6,14:IF flug%=1 THEN PRINT n$(4) E
LSE PRINT nt$(I1,4)
LOCATE 6,53:IF flug%=1 THEN PRINT n$(5) E
LSE PRINT nt$(I1,5)
LOCATE 8,39:IF flug%=1 THEN PRINT n$(6) E
LSE PRINT nt$(I1,6)
FOR Z=7 TO 25:LOCATE 9+Z-6,2:IF flug%=1 T
HEN PRINT n$(Z) ELSE PRINT nt$(I1,Z)
NEXT
IF flug%=1 AND FRE(0)>6000 AND I<massimo%
-10 THEN LOCATE 25,12:COLOR 3:PRINT "Vuoi
 trasferire il
 nominativo nel corrente archivio (S/N)?"
 ELSE LOCATE 25,23:COLOR 3:PRINT  "E' il
nominativo ric
hiesto (S/N)?"
RETURN

TRASFERISCI:
CLS:IF I=0 THEN COLOR 3:LOCATE 25,30:PRIN
T "L'archivio e' vuoto!!":FOR u=0 TO 1000
0:NEXT:CALL MA
SCHERA:RETURN
COLOR 3
n$(1)=" 1. Tutti i nominativi verso un Fi
le"
n$(2)=" 2. Selezione di nom. verso un Fil
e"
n$(3)=" 3. Ritorna al Menu principale"
```

```
n=3:stringa$="   Trasferimento:":GOSUB
MASCHERA3
IF p=1 THEN
  trasftutti:
  CLS:INPUT "Qual e' il File su cui trasf
erire il corrente archivio ";file$:IF fil
e$="" THEN tra
sftutti
  OPEN file$ FOR APPEND AS #1
  FOR I1=1 TO I
    FOR Z=1 TO 25
      PRINT#1,nt$(I1,Z)
    NEXT
  NEXT:CLOSE#1:GOTO TRASFERISCI
END IF
IF p=2 THEN
  Trasfselet:
  CLS:fleg%=0:I1=0:i2=0:INPUT "Qual e' il
 File destinazione ";file$:IF file$="" TH
EN Trasfselet
  OPEN file$ FOR APPEND AS #1
  loopfl4:
  i2=I1:flag%=0:GOSUB nome:IF flag%<>0 TH
EN i2=0:CLOSE#1:GOTO TRASFERISCI
  FOR Z=1 TO 25
    PRINT#1,nt$(I1,Z)
  NEXT
  fleg%=1:GOTO loopfl4
END IF
CALL MASCHERA:RETURN

SORT:
CLS:LOCATE 25,34:COLOR 3:PRINT "Attendere
!!!":GOSUB ORDINAMENTO:CALL MASCHERA:RETU
RN

' SUB INPUTC
' by Andrea De Prisco
' (C) 1987
'
SUB inputC(x,y,stringa$,lmin%,lmax%,tipo%
,scartaccetta$,codiciuscita$,cp) STATIC
IF codiciuscita$="" THEN codiciuscita$=CH
R$(13)
tasticursore$=CHR$(129)+CHR$(138)+CHR$(30
)+CHR$(31)
puntatore=LEN(stringa$)+1
stringa$=LEFT$(stringa$+SPACE$(lmax%),lma
x%)
IF cp=1 THEN LOCATE y,x+1:COLOR 2:PRINT s
tringa$:v%=y ELSE LOCATE y,x:COLOR 2:PRIN
T stringa$:v%=
y
getakey:
o%=x+puntatore:GOSUB accettacarattere
IF INSTR(codiciuscita$,car$)<>0 THEN
  comodo$=stringa$
  WHILE RIGHT$(stringa$,1)=" "
    stringa$=LEFT$(stringa$,LEN(stringa$)
-1)
  WEND
  IF LEN(stringa$)=>lmin% THEN EXIT SUB
  stringa$=comodo$
END IF
IF INSTR(tasticursore$,car$)<>0 THEN
  cur%=INSTR(tasticursore$,car$)
  IF cur%=1 THEN puntatore=1
  IF cur%=2 THEN puntatore=lmax%
  IF cur%=3 AND puntatore <lmax% THEN pun
tatore=puntatore+1
  IF cur%=4 AND puntatore>1 THEN puntator
e=puntatore-1
  GOTO getakey
END IF
IF codice%=8 AND puntatore>1 THEN
  stringa$=LEFT$(stringa$,puntatore-2)+RI
GHT$(stringa$,lmax%-puntatore+1)+" "
  GOSUB stampastringa:puntatore=puntatore
-1:GOTO getakey
END IF
IF codice%=127 THEN
  stringa$=LEFT$(stringa$,puntatore-1)+RI
GHT$(stringa$,lmax%-puntatore)+" "
  GOSUB stampastringa:GOTO getakey
END IF
IF codice%=27 THEN
  stringa$=SPACE$(lmax%):GOSUB stampastri
nga:puntatore=1
  SOUND 2000,2,150,0:GOTO getakey
END IF
GOSUB controllocarattere
IF flag%=0 AND puntatore<=lmax% THEN
  stringa$=LEFT$(stringa$,puntatore-1)+ca
r$+RIGHT$(stringa$,lmax%-puntatore+1)
  stringa$=LEFT$(stringa$,lmax%)
  GOSUB stampastringa:puntatore=puntatore
+1:IF puntatore>lmax% THEN puntatore=punt
atore-1
  ELSE
    BEEP
END IF
GOTO getakey
stampastringa:
LOCATE y,x+1:PRINT stringa$
RETURN
controllocarattere:
flag%=0
IF tipo%<6 THEN IF INSTR(scartaccetta$,ca
r$)<>0 THEN flag%=1:RETURN
IF tipo%=0 AND codice%>31 AND codice%<127
 THEN RETURN
IF tipo%=1 OR tipo%=2 THEN
  IF codice%<>32 AND NOT (codice%>64 AND
codice%<91) AND NOT (codice%>96 AND codic
e%<123) THEN f
 lag%=-1
  IF tipo%=2 THEN RETURN
  IF codice%>47 AND codice%<58 THEN flag%
=0
  RETURN
END IF
IF tipo%=3 OR tipo%=4 OR tipo%=5 THEN
  IF (codice%<43 OR codice%>57) OR codice
%=44 OR codice%=47 THEN flag%=-1
  IF (codice%=43 OR codice%=45) AND (punt
atore<>1 OR LEFT$(stringa$,1)="+" OR LEFT
$(stringa$,1)=
"-") THEN flag%=-1
  IF (codice%>47 AND codice%<58) AND INST
R("+-",LEFT$(stringa$,1))<>0 AND puntator
e=1 THEN flag%
=-1
  IF tipo%=3 THEN RETURN
  IF codice%=46 THEN flag%=-1
  IF tipo%=4 THEN RETURN
  IF codice%=43 OR codice%=45 THEN flag%=
-1
  RETURN
END IF
IF INSTR(scartaccetta$,car$)=0 THEN flag%
=-1
RETURN
accettacarattere:
flipflop=1
copiadicar$=MID$(stringa$,puntatore,1)
Stampa:
LOCATE v%,o%
IF flipflop=1 THEN PRINT "_" ELSE PRINT c
opiadicar$
contatore=1
loop:
car$=INKEY$
IF car$<>"" THEN
  codice%=ASC(car$)
  IF codice%>96 AND codice%<123 THEN codi
ce%=(codice%-32)
  car$=CHR$(codice%)
  LOCATE v%,o%:PRINT copiadicar$:RETURN
END IF
contatore=contatore+1:IF contatore>60 THE
N flipflop=3-flipflop:GOTO Stampa ELSE GO
TO loop
END SUB

SUB MASCHERA STATIC
SHARED I,Arch$,massimo%
```

```
CLS
COLOR 2,0:IF Arch$<>"Archivio" THEN LOCAT
E 5,INT(16-(LEN(Arch$))/2) ELSE LOCATE 5,
15
IF Arch$<>"Archivio" THEN PRINT "Nell'arc
hivio ";:COLOR 1:PRINT Arch$;:COLOR 2:PRI
NT " sono cont
enuti";I;"nominativi." ELSE PRINT "Nell'a
rchivio sono contenuti ";I;" nominativi."
IF FRE(0)<=6000 OR I>=massimo%-10 THEN LO
CATE 7,27:COLOR 3:PRINT "Memoria Esaurita
!!" ELSE LOCAT
E 7,21:COLOR 3:PRINT "Sono disponibili ";
FRE(0)-6000;" bytes."
LOCATE 10,1:COLOR 2,0
PRINT TAB(20)"         Gestione Scambi"
PRINT
PRINT TAB(20)"    written by M.Costantin
o"
PRINT TAB(20)" Pubblicato da Commodore Ga
zette"
LINE (142,67)-(432,109),1,b
LINE (140,66)-(434,110),1,b
LINE (138,65)-(436,111),2,b
LINE (136,64)-(438,112),2,b
LINE (134,63)-(440,113),3,b
LINE (132,62)-(442,114),3,b
END SUB

SUB MASCHERA2 STATIC
SHARED I
CLS
LINE (8,3)-(608,36),1,b
LINE (8,20)-(608,20),1
LINE (360,3)-(360,20),1
LINE (8,36)-(608,52),1,b
LINE (8,52)-(608,68),1,b
LINE (292,52)-(292,68),1
LINE (6,228)-(610,228),1
LOCATE 2,3:COLOR 3:PRINT "Cognome:"
LOCATE 2,47:PRINT "Nome:"
LOCATE 4,3:PRINT "Indirizzo:"
LOCATE 6,3:PRINT "Citta':"
LOCATE 6,47:PRINT "Tel."
LOCATE 8,30:PRINT "Scheda"
LOCATE 30,17:COLOR 2:PRINT "Nell'archivio
 sono contenuti "I" nominativi.";:COLOR 3
END SUB

ORDINAMENTO:
IF I=0 OR I=1 THEN RETURN
FOR i3=1 TO I-1
f=0
FOR k=1 TO I-1
LOCATE 1,1:PRINT i3;", ";k
IF ASC(nt$(k+1,1))=27 OR ASC(nt$(k,1))=27
 THEN riorganizza
IF ASC(nt$(k+1,1))=>ASC(nt$(k,1)) THEN 14
FOR Z=1 TO 25:SWAP nt$(k+1,Z),nt$(k,Z):NE
XT
f=1
14 NEXT
IF f=0 THEN RETURN
NEXT:RETURN
riorganizza:
IF ASC(nt$(k,1))=27 THEN k=k-1
FOR k3=k+1 TO I
FOR Z=1 TO 25:nt$(k3,Z)=nt$(k3+1,Z):NEXT
NEXT
I=I-1
FOR Z=1 TO 25
  nt$(I+1,Z)=""
NEXT
RETURN

chiamaI:
IF imm$<>"" THEN cp=1
scarta$=",:;"+CHR$(34):CALL inputC(o,v,im
m$,a%,b%,tipo%,scarta$,cod$,cp)
RETURN

SALVA:
```

```
CLOSE#1
OPEN Arch$ FOR OUTPUT AS #1
FOR I1=1 TO I
  FOR Z=1 TO 25
    PRINT#1,nt$(I1,Z)
  NEXT
NEXT:CLOSE#1:RETURN

MASCHERA3:
COLOR 0,0:CLS:k=0:u=0:p=0:a=0:b=0:c=MOUSE
(0):RESTORE
k=6
LINE (128,k+6)-(457,k+18),1,bf
LINE (115,k)-(445,k+12),2,bf
LOCATE 2,17:COLOR 0,2:PRINT stringa$
IF n>8 THEN n=8
FOR u=1 TO n
k=k+24
LINE (128,k+6)-(457,k+18),1,bf
LINE (115,k)-(445,k+12),2,bf
READ a:LOCATE a,17
DATA 5,8,11,14,17,20,23,26,29
COLOR 0,2:PRINT n$(u)

NEXT
k=k+48
LINE (128,k+6)-(457,k+18),1,bf
LINE (115,k)-(445,k+12),2,bf
LOCATE a+6,17:COLOR 1,2:PRINT "Utilizza i
l Mouse per scegliere"
MOUSE ON
166 RESTORE 165:READ a,b
WHILE MOUSE(0)=0 OR MOUSE(0)=1 OR MOUSE(0
)=2 OR MOUSE(0)=3 OR MOUSE(0)=-1 OR MOUSE
(0)=-2
WEND
y=MOUSE(0)
currentX=MOUSE(1)
currentY=MOUSE(2)
FOR u=1 TO n
IF currentX>128 AND currentX<457 AND curr
entY>a AND currentY<b THEN p=u:COLOR 1,0:
MOUSE OFF:RETU
RN
165 DATA 30,48,54,72,78,96,102,120,126,14
4,150,168,174,192,198,216,222,240,246,264
READ a,b
NEXT
GOTO 166

start1:
SCREEN 1,640,512,2,4
WINDOW 1,"Gestione Scambi - Written by Ma
rco Costantino - Commodore Gazette - Rev
3.41",(0,1)-(6
30,497),0,1
PALETTE 0,0,0,0:PALETTE 1,1,0,0
PALETTE 2,1,1,0:PALETTE 3,0,1,0
RETURN

SUB maschera4 STATIC
SHARED I
CLS
LINE (8,3)-(608,36),1,b
LINE (8,20)-(608,20),1
LOCATE 2,17:COLOR 2:PRINT  "Nell'archivio
 sono contenuti "I" nominativi.";:COLOR 3
LOCATE 4,3:COLOR 3:PRINT "N."
LINE (8,3)-(8,512),1
LINE (60,20)-(60,484),1
LOCATE 4,10:COLOR 3:PRINT "CODICE"
LINE (132,20)-(132,484),1
LOCATE 4,19:COLOR 3:PRINT "COGNOME"
LINE (276,20)-(276,484),1
LOCATE 4,37:COLOR 3:PRINT "NOME"
LINE (420,20)-(420,484),1
LOCATE 4,55:COLOR 3:PRINT "CITTA'"
LINE (608,20)-(608,512),1
LINE (8,484)-(608,484),1
LOCATE 62,15:COLOR 2:PRINT "Premi un tast
o per continuare, ESC per uscire";:LOCATE
 1,1
END SUB
```

I drive ottici: trenta hard disk da 20 MB in un CD ROM

Perché non esiste ancora un drive CD ROM per l'Amiga? La situazione statunitense per quanto riguarda diffusione e assorbimento delle nuove tecnologie laser

di Matthew Leeds

Io ricevo moltissima posta. In genere si tratta di inutile spazzatura che offre sconti per acquisti e servizi che non mi interessano, tagliandi per prenotazione di merce che possiedo già, riviste, copie saggio di software, comunicati stampa... capite anche voi a che cosa mi riferisco. Ma talvolta mi arrivano anche lettere che riguardano gli articoli che scrivo, e a quelle cerco sempre di rispondere.

Recentemente ho ricevuto una lettera che allegava un ritaglio tratto da una *newsletter* di un club di utenti Atari. Nel ritaglio era citata l'imminente disponibilità di un drive CD ROM per i computer Atari a 16 bit, e la lettera parlava del fatto che non esistono prodotti simili per l'Amiga. Ah, dimenticavo, il prezzo previsto per il drive CD ROM dell'Atari era di 650 dollari.

Ho cercato di verificare la notizia presso parecchi computer shop locali e non ho trovato la minima traccia di un drive CD ROM Atari in vendita. Naturalmente l'annuncio di un prodotto non comporta necessariamente la sua immediata disponibilità, lo testimoniano innumerevoli casi compreso il mai realizzato computer LCD della Commodore.

Ma questa lettera mi ha comunque dato lo spunto per fare una panoramica sull'attuale tecnologia dei dischi ottici, con lo sguardo sempre rivolto al loro uso per l'Amiga.

Dal punto di vista teorico, un CD ROM è un oggetto molto semplice. La tecnologia di base è quella dei CD audio, con la differenza che le informazioni immagazzinate in un CD ROM non sono destinate alla riproduzione audio ma all'uso da parte di un computer. Di conseguenza il livello di fedeltà richiesto è molto più alto.

Benché la proliferazione di lettori di CD audio abbia ridotto anche i costi di produzione per i lettori di CD ROM, è inevitabile che il loro costo unitario resti sempre più alto a causa dell'esigenza di dispositivi hardware aggiuntivi per garantire l'integrità dei dati. Per non parlare dell'incidenza sui costi di una produzione ancora limitata o dell'evidente differenza di valore tra un lettore di CD audio e di CD ROM.

Nell'intelligente tentativo di evitare le difficoltà di formato che si presentano a causa dell'attuale differenziazione del software (avete mai provato a gestire informazioni scritte per una certa macchina usandone un'altra? Per esempio a mettere d'accordo il Mac e l'Amiga?) è subito stato previsto uno standard per i dati CD ROM. Viene chiamato standard "High Sierra". Quasi tutti i titoli disponibili su CD ROM seguono questo formato e tutti i lettori che saranno eventualmente prodotti per l'Amiga dovranno senza dubbio adeguarsi allo stesso standard.

Il costo dei lettori ora esistenti, destinati ad altre macchine, si aggira intorno ai mille dollari. La maggior parte dei produttori offre sconti sui lettori CD ROM quando vengono acquistati assieme ai loro prodotti, e non c'è da stupirsene, dal momento che alcuni database su CD ROM hanno prezzi sull'ordine delle decine di migliaia di dollari.

I compact disk ROM offrono molteplici vantaggi rispetto ad altre forme di immagazzinamento e distribuzione delle informazioni. Garantiscono prima di tutto un notevole risparmio di costi in alcune applicazioni. Hanno una capacità media di memoria di 550 MB. Gli attuali costi di produzione (trascurando la preparazione dei dati, il cui costo non cambia) vanno dai 5 ai 50 dollari per disco, a seconda del numero di copie prodotte. Nel caso di

produzione di massa, il costo per MByte scende a 0,009 dollari: anche calcolando il profitto dei produttori il prezzo per MByte resterà di gran lunga il più basso di qualsiasi altro supporto per la gestione di dati.

Un altro vantaggio è la sicurezza che i propri dati rimarranno integri praticamente per sempre. I CD conservano la loro efficienza molto a lungo, è difficile cancellarli per errore, tollerano nel modo migliore i trattamenti rudi e con loro non c'è alcun pericolo di rottura della testina o formattazione accidentale.

I CD ROM sono anche facili da trasportare: pensate a cosa significherebbe portare con sé 550 MB di dati su floppy disk, cioè almeno 600 dischi da 3,5" o 27 hard disk da 20 MB... sarebbe un po' difficile metterseli in tasca!

Inoltre i CD ROM non sono facilmente "pirateggiabili"; certo, è possibile copiarli su dischi normali, ma ne occorrerebbero davvero parecchi: non ci si può limitare a digitare DISKCOPY e aspettare.

Ma se presentano tanti vantaggi, perché non sono ancora disponibili per il mercato degli home e personal computer? Ci sono diversi motivi. Vediamoli uno per uno.

1) Sono pochissimi i consumatori che hanno bisogno di una così grande quantità di informazioni in forma leggibile dal proprio computer. A parte le enciclopedie, il mercato non propone prodotti per i quali siano necessarie le potenzialità di un CD ROM. A voi interessa tutta la Biblioteca del Congresso a casa vostra su compact?

2) Il costo di un "master" è molto alto. Una cosa è spendere uno o due anni di lavoro per creare un gioco o un word processor, un'altra è creare 550 MB di informazioni utili. Una volta di più, la grande massa delle applicazioni previste per il CD ROM è destinata al mondo degli affari o dell'industria.

3) Un lettore adatto a un CD ROM ha un costo ancora improponibile per il mercato degli home e personal computer. Anche per quanto riguarda i drive per gli hard disk, la maggior parte dei proprietari di home computer ha affrontato l'acquisto solo quando il prezzo è sceso molto al di sotto dei 1.500 dollari, e fino a pochi anni fa il prezzo di un drive per hard disk da 20 MB era questo.

4) La sindrome dell'uovo e della gallina. Il mercato dei prodotti CD ROM, al momento attuale, è solo potenziale e i fabbricanti sono riluttanti a iniziare una vasta produzione di CD ROM finché non esisterà una solida base di consumatori (vale a dire fino a quando voi e io non avremo acquistato un lettore). Ma voi siete riluttanti (e anch'io) ad acquistarne uno finché non ne sentirete il bisogno, vale a dire fino a quando non esisterà sul mercato un programma su CD ROM che vi interessa. Ecco che si instaura un circolo vizioso nel quale nessuno (consumatore e produttore) vuole affrontare rischi.

E allora dove possono trovare collocazione i CD ROM? Applicazioni tecnologiche e scientifiche come il CAD/CAM, lo studio delle immagini generate via satellite, e l'ingegneria architettonica hanno dimostrato un grande bisogno di enormi capacità di immagazzinare dati. A loro sono destinati database contenenti rappresentazioni grafiche, copie complete di giornali, riviste, atti di convegni e così via.

Il mercato legale è stato una miniera. Cause, statuti, precedenti, registrazioni, concessioni e marchi di fabbrica, regolamenti statali e federali, e tutta la legislazione sulle imposte sono disponibili su CD ROM e le informazioni vengono regolarmente aggiornate.

Un altro campo interessante è quello medico. I testi di consultazione di medicina richiedono un'enorme quantità di dati in forma grafica, riferimenti incrociati e grande accuratezza. Database specializzati sono stati destinati a campi molto specifici come farmacologia, tossicologia, indennità assicurative, e tecniche di pronto soccorso.

Il governo è uno dei maggiori utenti di CD ROM. I militari hanno manuali per ogni cosa: mantenere una documentazione di tutto su CD ROM è un obiettivo molto meno irraggiungibile che tenere in magazzino tutte le copie necessarie a rifornire ogni base che potrebbe farne richiesta. Il costo effettivo di questa applicazione è il puro e semplice immagazzinamento dei dati tramite mezzi ottici.

Finché i CD ROM verranno prodotti esclusivamente per applicazioni come queste, credo che un lettore per CD ROM destinato all'Amiga sia impensabile. D'altra parte c'è un altro mezzo ottico disponibile, il WORM (Write Once Read Many, scrivi una volta leggi molte volte), che offre una capacità di immagazzinare scrittura ottica di circa 400 MB in una cartuccia estraibile. Sia la C Ltd sia la CSA hanno annunciato soluzioni tecnologiche che offrono anche ai possessori dell'Amiga la possibilità di accedere a questo tipo di dispositivo.

Mi occuperò del drive WORM in uno dei prossimi articoli. Intanto, se l'argomento vi appassiona e siete curiosi di sapere qualcosa di più sui CD ROM, anche dal punto di vista tecnico, vi ricordo che su *Commodore Gazette* n. 4/86 è stato pubblicato un interessante articolo ("La memoria di massa e i compact disk"), che metteva a confronto floppy disk e tecnologie laser. ■

Per ulteriori informazioni contattare direttamente:

C Ltd
723 East Skimer
Wichita, Kansas 67211
USA

CSA
7564 Trade Street
San Diego, CA 92121
USA

CORSO DI PROGRAMMAZIONE IN LM PER C-64: QUATTORDICESIMA PUNTATA

LE ANIMAZIONI E GLI INTERRUPT DEL VIC

Gli sprite, strumenti semplici ed efficaci per creare animazioni sullo schermo. Il raster, le sue funzioni e come farne uso. Lo "scroll fine" dello schermo. Gli interrupt prodotti dal VIC per le collisioni degli sprite e per il raster

di Sergio Fiorentini

(N.d.r.) Nel corso dell'articolo l'autore ha talvolta utilizzato lettere maiuscole in modo sintatticamente improprio, per particolari esigenze didattiche. Poiché le puntate che costituiscono questo corso si integrano reciprocamente, consigliamo a chi ci segue per la prima volta di richiedere le copie arretrate della rivista (numeri 1-2-3-4/86, 1-2-3-4-5-6-7/87, 1-2/88).

Dopo aver esaminato, nelle due precedenti puntate, i diversi modi grafici per la visualizzazione dei testi e delle immagini, analizziamo in questa lezione le capacità d'animazione del C-64.

Le animazioni

Le animazioni (definite "sprite", in inglese) sono elementi grafici simili ai caratteri programmabili, dai quali si differenziano però in alcuni aspetti fondamentali: possono essere più grandi ed essere collocate in qualsiasi punto dello schermo.

Dal momento che la posizione di uno sprite sullo schermo è indipendente dalla sua collocazione in memoria (cosa che ne semplifica e ne accelera incredibilmente la gestione), per muoverlo è sufficiente variare il contenuto di due registri presenti nel VIC, creando così i tipici effetti d'animazione comuni a molti giochi. Gli sprite, inoltre, sono completamente indipendenti dal tipo di modo grafico attivato per lo schermo.

Queste caratteristiche sono fondamentali nella creazione delle animazioni, e vengono ottenute grazie a una particolare configurazione hardware. Arrivare agli stessi risultati via software sarebbe praticamente impossibile.

L'interfaccia video del C-64 è in grado di gestire sino a otto animazioni, ognuna delle quali possiede una propria forma, un proprio colore (e anche più d'uno, se ci troviamo in modo multicolor), e una propria posizione sullo schermo, indipendente da ogni altra animazione attivata.

Infine, ogni animazione viene individuata da un numero (da 0 a 7) che permette di distinguerla da tutte le altre.

I modi e i colori delle animazioni

Le animazioni, analogamente ai caratteri programmabili, hanno a disposizione due modi grafici per essere rappresentate: alta risoluzione e multicolor.

Tramite i bit del registro $D01C (selezione animazioni multicolor) si stabilisce quali devono essere le animazioni in alta risoluzione e quali in multicolor.

Per l'esattezza, a ogni bit del registro è associata la corrispondente animazione (bit 0 – animazione 0, bit 1 – animazione 1...); accedendo quindi a ogni singolo bit è possibile stabilire il modo grafico del corrispondente sprite.

Le animazioni in alta risoluzione sono costituite da 504 pixel articolati in 21 linee per 24 colonne, per un totale di 63 byte. A questi si aggiunge un byte che permette di disporre gli sprite in memoria secondo indirizzi che sono tutti potenze di 2. Ogni sprite impegna quindi un'area di memoria da 64 byte, che ne definisce l'aspetto.

I primi 3 byte di quest'area (matrice dell'animazione) corrispondono ai pixel della prima

ANIMAZIONE

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

COLORE TRASPARENTE

COLORE ANIMAZIONE

MULTICOLOR 0

MULTICOLOR 1

MATRICE CORRISPONDENTE

%00000000	%00000000	%00000000
%00000000	%00000000	%00000000
%00111111	%11111111	%11111100
%00111111	%11111111	%11111100
%00111110	%10101010	%10111100
%00111110	%10101010	%10111100
%00111110	%10010110	%10111100
%00111110	%10010110	%10111100
%00111110	%10010110	%10111100
%00111110	%10010110	%10111100
%00111110	%10010110	%10111100
%00111110	%10010110	%10111100
%00111110	%10010110	%10111100
%00111110	%10010110	%10111100
%00111110	%10010110	%10111100
%00111110	%10010110	%10111100
%00111110	%10101010	%10111100
%00111111	%11111111	%11111100
%00111111	%11111111	%11111100
%00000000	%00000000	%00000000
%00000000	%00000000	%00000000

ne e la corrispondente matrice nel caso che questa sia multicolor.

Poiché il VIC, per poter visualizzare un'animazione sullo schermo, deve accedere alla corrispondente matrice, questa deve trovarsi nel banco video correntemente visualizzato.

Inoltre, poiché l'intero banco video si estende per 16K e ogni matrice occupa 64 byte, è possibile allocare i dati grafici degli sprite in 256 posizioni diverse (16384/64 = 256), associando a ogni matrice un numero compreso tra 0 e 255 detto *numero della matrice*. Per comunicare al VIC l'indirizzo al quale reperire la matrice di uno sprite, è sufficiente memorizzare il numero della matrice in una delle ultime 8 locazioni puntatore presenti dopo la memoria di schermo. Per esempio, allo sprite 0 possiamo assegnare ora la matrice 155 ora la matrice 228, semplicemente memorizzando questi due numeri nell'apposita locazione puntatore.

Per calcolare l'indirizzo della locazione puntatore assegnata all'ennesimo sprite, si può utilizzare la seguente formula:

Locazione puntatore animazione = Base memoria schermo + $03F8 + Numero animazione (da 0 a 7)

A titolo d'esempio, ponendo che la memoria di schermo sia allocata nella posizione standard (a partire da $0400, nel banco video 0), e volendo che lo sprite 3 visualizzi la matrice di 64 byte memorizzata a partire dall'indirizzo $2000, occorrerà scrivere il numero della matrice $80 ($2000/64) nella locazione della memoria di schermo $07FB = $0400+$03F8+$3.

Sebbene un banco video possa contenere sino a 256 matrici di animazioni, in pratica questo numero è molto inferiore, poiché all'interno del banco video risiedono anche la memoria di schermo e le eventuali memorie caratteri o "mappe di bit", necessarie alla visualizzazione dello sfondo su cui si muovono le animazioni.

Una volta che la matrice d'animazione è stata collocata in memoria ed è stato impostato il relativo puntatore nella memoria di schermo, per attivarne la visualizzazione è necessario "accenderla" impostando a %1 il corrispondente bit del registro $D015. In questo registro ogni bit individua lo stato di visibile/non visibile di uno sprite.

Il segmento di programma che segue consente di visualizzare un'animazione in alta risoluzione utilizzando lo sprite 0.

```
         .ORG  $1000                 ;Inizio programma
BasSch   .EQU  $0400                 ;Base memoria schermo
NumAni   .EQU  0                     ;Numero animazione
ColAni   .EQU  $D027                 ;Registro colore
                                     ;animazione 0
Rosso    .EQU  2                     ;Codice colore rosso
AttAni   .EQU  $D015                 ;Attivazione animazioni
PosXAni  .EQU  $D000+(NumAni*2) ;Posizione orizzontale
                                     ;animazione 0
```

```
PosYAni  .EQU  $D001+(NumAni*2) ;Posizione verticale
                                ;animazione 0
BaseMat  .EQU  $2000            ;Inizio matrice animazione
NumMat   .EQU  BaseMAT/64       ;Numero matrice
         LDX   #63              ;Copia nella RAM la matrice
                                ;contenuta nel programma
Ripeti   LDA   MAT,X            ;stesso a partire
         STA   BaseMat,X        ;dall'indirizzo:BaseMat
         DEX
         BPL   Ripeti

         LDA   #Rosso           ;Colore animazione 0
         STA   ColAni+NumAni    ;uguale rosso
         LDA   #100             ;Posiziona l'animazione
                                ;0 sullo schermo,
         STA   PosXAni          ;in modo che
                                ;non sia coperta
         STA   PosYAni          ;dal bordo

         LDA   AttAni           ;Attiva l'animazione 0
         ORA   #1<NumAni        ;impostando il bit 0 nel
         STA   AttAni           ;registro AttAni

         RTS                    ;termine programma

                                ;Definizione della matrice
                                ;impiegata per
                                ;l'animazione

MAT      .BYTE %00000000,%00000000,%00000000
         .BYTE %00000000,%00000000,%00000000
         .BYTE %00111111,%11111111,%11111100
         .BYTE %00111111,%11111111,%11111100
         .BYTE %00111100,%00000000,%00111100
         .BYTE %00111100,%00000000,%00111100
         .BYTE %00111100,%00000000,%00111100
         .BYTE %00111100,%00000000,%00111100
         .BYTE %00111100,%00000000,%00111100
         .BYTE %00111100,%00000000,%00111100
         .BYTE %00111100,%00000000,%00111100
         .BYTE %00111100,%00000000,%00111100
         .BYTE %00111100,%00000000,%00111100
         .BYTE %00111100,%00000000,%00111100
         .BYTE %00111100,%00000000,%00111100
         .BYTE %00111100,%00000000,%00111100
         .BYTE %00111100,%00000000,%00111100
         .BYTE %00111111,%11111111,%11111100
         .BYTE %00111111,%11111111,%11111100
         .BYTE %00000000,%00000000,%00000000
         .BYTE %00000000,%00000000,%00000000
         .BYTE %00000000

         .END
```

Nel programma precedente, per determinare la posizione dello sprite sono stati impostati i due registri PosXAni e PosYAni, che costituiscono le coordinate di schermo dell'animazione.

Poiché il centro di queste coordinate si trova sotto il bordo nell'angolo superiore sinistro dello schermo, se queste fossero state impostate a 0, l'animazione sarebbe stata interamente occultata dal bordo.

Ingrandimento delle animazioni

Tutte le otto animazioni possono essere ingrandite sia orizzontalmente sia verticalmente, raddoppiandone larghezza e altezza.

Il registro $D017 contiene i bit che controllano le dimensioni verticali di ogni animazione; se uno di questi bit è impostato a %1, la corrispondente animazione viene visualizzata raddoppiata in altezza. Analogamente, il registro $D01D contiene i bit che controllano le dimensioni orizzontali delle otto animazioni. In questo caso un bit impostato a %1 ha come effetto la visualizzazione dell'animazione corrispondente con la larghezza raddoppiata.

Il segmento di codice che segue, da inserirsi nel programma precedente prima dell'istruzione RTS, determina l'ingrandimento verticale e orizzontale dell'animazione 0.

```
IngXAni  .EQU  $D01D            ;Registro ingrandimento
                                ;Ani. X
IngYAni  .EQU  $D017            ;Registro ingrandimento
                                ;Ani. Y

         LDA   IngXAni          ;Impostazione
         ORA   #1<NumAni        ;ingrandimento
                                ;orizzontale
         STA   IngXAni
         LDA   IngYAni          ;Impostazione
         ORA   #1<NumAni        ;ingrandimento
                                ;verticale
         STA   IngYAni
```

Collocazione delle animazioni

Una caratteristica peculiare delle animazioni è la possibilità di situarle in qualsiasi punto dello schermo, inserendo semplicemente le opportune coordinate nei due appositi registri del VIC.

Le coordinate di schermo impiegate con gli sprite adottano il pixel come unità di misura, a prescindere dal modo grafico attivato sullo schermo. L'origine (punto di ascissa 0 e ordinata 0) è situata nel margine in alto a sinistra dello schermo, sotto il bordo.

Le coordinate di un'animazione si riferiscono al suo angolo superiore sinistro, e sono indipendenti sia dal modo grafico utilizzato per visualizzarla, sia dall'eventuale ingrandimento verticale e orizzontale.

Dal momento che l'origine delle coordinate

esterna allo schermo visibile, il primo pixel visibile in alto a sinistra ha coordinate (24, 50) ($18, $32). Se un'animazione viene collocata sullo schermo con queste coordinate, risulta interamente visibile.

Dal momento che lo schermo si estende in larghezza per 320 ($140) pixel e in altezza per 200 ($C8) pixel, l'ultimo punto visualizzabile in basso a destra è individuato dalle coordinate (343, 249) ($157, $FA).

Vediamo come vengono rappresentate in memoria le coordinate di un'animazione. L'ordinata è interamente rappresentabile tramite un byte. È quindi sufficiente un registro da 1 byte, detto PosY, per individuarla.

Per quanto invece riguarda l'ascissa di un'animazione, il discorso è leggermente più complesso in quanto la sua rappresentazione numerica ha bisogno di 9 bit, e quindi 1 byte è insufficiente per individuarla.

Per questa ragione, gli 8 bit meno significativi di ogni coordinata orizzontale sono memorizzati in un byte registro (PosX), assegnato a ogni sprite, mentre tutti i bit 9 delle ascisse delle 8 animazioni sono raccolti nello stesso byte, detto registro MSBPosX.

In questo registro i bit 9 di ogni sprite sono distribuiti nel consueto ordine: il bit 0 si riferisce all'animazione 0, il bit 1 all'animazione 1, e così via.

Ecco ora la tavola di tutti i registri interessati alla collocazione delle animazioni sullo schermo.

Registro	Indirizzo	Descrizione
PosX0	$D000	Posizione orizzontale Ani. 0
PosY0	$D001	Posizione verticale Ani. 0
PosX1	$D002	Posizione orizzontale Ani. 1
PosY1	$D003	Posizione verticale Ani. 1
PosX2	$D004	Posizione orizzontale Ani. 2
PosY2	$D005	Posizione verticale Ani. 2
PosX3	$D006	Posizione orizzontale Ani. 3
PosY3	$D007	Posizione verticale Ani. 3
PosX4	$D008	Posizione orizzontale Ani. 4
PosY4	$D009	Posizione verticale Ani. 4
PosX5	$D00A	Posizione orizzontale Ani. 5
PosY5	$D00B	Posizione verticale Ani. 5
PosX6	$D00C	Posizione orizzontale Ani. 6
PosY6	$D00D	Posizione verticale Ani. 6
PosX7	$D00E	Posizione orizzontale Ani. 7
PosY7	$D00F	Posizione verticale Ani. 7
MSBPosX	$D010	Bit piu' significativo delle coordinate orizzontali

Il segmento di codice che segue, che dev'essere inserito (nel programma già presentato) prima dell'RTS, consente di muovere orizzontalmente l'animazione 0.

```
MSBPosX  .EQU  $D010   ;Bit piu' significativo delle
                       ;ascisse
RIT      .EQU  $10     ;Costante di ritardo
Ripeti   LDX   #RIT    ;Realizza un ciclo di ritardo
         LDY   #0      ;di alcuni secondi
                       ;Per accelerare
Ripeti2  DEY           ;l'animazione si deve
                       ;diminuire la
         BNE   Ripeti2 ;quantita' RIT
         DEX
         BNE   Ripeti2

         INC   PosX0   ;incrementa la posizione
         BNE   Ripeti  ;orizzontale dell'animazione e,
         LDA   MSBPosX ;se necessario, complementa il
         EOR   #$01    ;bit piu' significativo
                       ;(il nono),
         STA   MSBPosX ;quindi ritorna all'esecuzione
         CLC           ;del ciclo di ritardo
         BCC   Ripeti
```

Priorità delle animazioni

Quando due o più animazioni si sovrappongono sullo schermo, quella assegnata al numero d'identificazione più basso ha una priorità di schermo maggiore, ovvero appare sopra tutte le altre animazioni, dando l'idea di essere più vicina all'osservatore. Se per esempio lo sprite 0 viene visualizzato nella stessa posizione dello sprite 1, quest'ultimo viene coperto dal primo. La priorità delle animazioni, ovvero l'ordine con cui le animazioni si sovrappongono una sull'altra, non è modificabile.

La priorità animazione-carattere può essere invece programmata per ogni singolo sprite, facendo in modo che lo sprite appaia sopra o sotto i caratteri (o la grafica) costituenti lo sfondo.

A questo scopo è stato previsto il registro $D01B i cui singoli bit controllano la priorità tra lo sfondo e le animazioni. Se un bit è impostato a %1, l'animazione che gli corrisponde appare "sopra" i caratteri dello sfondo; viceversa, se questo bit è impostato a %0, l'animazione viene visualizzata "sopra" il colore di fondo dello schermo ma "sotto" i caratteri costituenti lo sfondo.

Lasciamo al lettore l'onere d'inserire le istruzioni necessarie per variare la priorità animazione-carattere nel programma che abbiamo proposto.

Collisioni delle animazioni

Quando due animazioni si toccano, ovvero quando due pixel non trasparenti di sprite diversi si sovrappongono sullo schermo, in uno s...

registro (all'indirizzo $D01E) vengono impostati a %1 i bit corrispondenti alle animazioni interessate al contatto.

In modo analogo, un altro registro (all'indirizzo $D01F) fornisce indicazione dell'eventuale collisione fra le animazioni e i caratteri dello sfondo. Anche in questo caso verranno impostati a %1 i bit corrispondenti alle animazioni entrate in collisione con i caratteri dello sfondo.

È necessario prestare attenzione al fatto che i pixel multicolor dello schermo (caratteri o bitmap) corrispondenti alla sequenza %01, vengono considerati alla stessa stregua del colore di fondo, ovvero non originano collisioni con le animazioni.

I due registri di segnalazione delle collisioni hanno la caratteristica d'impostarsi nuovamente a $00 ogni volta che vengono letti.

Contatore di scansione

Come avevamo visto in una delle precedenti puntate, l'immagine sullo schermo televisivo o sul monitor si ottiene tramite un pennello elettronico che percorre l'intero schermo seguendo linee orizzontali e procedendo dall'alto verso il basso.

All'interno dell'interfaccia video esiste un apposito registro, detto "raster register", che viene continuamente aggiornato con il valore della linea di schermo che il pennello elettronico sta scorrendo. Questo registro viene quindi azzerato al termine di ogni quadro e incrementato a ogni nuova linea di scansione visualizzata sullo schermo. Durante la visualizzazione della prima linea del bordo, linea che potrebbe anche non essere visibile essendo rappresentata nella parte più alta dello schermo, il raster register contiene il valore $00; durante la scansione della seconda linea contiene il valore $01, e così via fino a coprire l'intero schermo.

Benché l'area indirizzabile dello schermo, cioè quella visibile non coperta dal bordo, si estenda dalla linea numero 50 ($32) alla linea 250 ($FA), il contatore di quadro si estende per 9 bit. Infatti durante la visualizzazione di gran parte del bordo inferiore il valore di questo contatore è superiore a $FF, massima cifra rappresentabile con un solo byte.

La parte meno significativa del raster register può essere letta accedendo alla locazione $D012, mentre il bit più significativo di questo contatore è il bit 7 del registro di controllo situato all'indirizzo $D011.

Leggendo ciclicamente il valore del raster register è possibile dividere lo schermo in due finestre con caratteristiche diverse: la parte superiore può, per esempio, essere gestita in modo grafico, mentre la parte inferiore in modo testo.

Utilizzando sempre lo stesso sistema è possibile visualizzare contemporaneamente sullo schermo più di 8 sprite: in questo caso, raggiunta una determinata linea di scansione, i puntatori agli sprite contenuti negli ultimi otto byte della memoria di schermo vengono modificati in modo che puntino a nuovi sprite. Analogamente, vengono modificati anche i vari attributi degli sprite, come la posizione, il colore, le dimensioni... Ovviamente, perché l'operazione abbia successo, nessuno sprite deve trovarsi sulla linea di divisione delle due finestre.

L'esempio che segue illustra semplicemente come cambiare il colore del bordo ogni 16 linee di scansione.

```
            .ORG $1000
Col.bordo   .EQU $D020          ;colore del bordo
Cont.linea  .EQU $D012          ;contatore di linea

Ripeti      LDA  Cont.linea     ;legge il contatore di linea
            AND  #$0F           ;e ogni 16 linee incrementa
            BNE  Ripeti         ;il colore del bordo
            INC  Col.bordo
            CLC
            BCC  Ripeti

.END
```

Il programma precedente, come del resto tutti i programmi che leggono il contatore di linea, ha il difetto di dover aspettare che il contatore di linea eguagli un certo valore di confronto, e di non permettere alla CPU di svolgere altri compiti nel frattempo. Si possono evitare questi sprechi di tempo facendo in modo che il microprocessore venga "avvertito" dall'interfaccia video non appena il contatore di linea ha raggiunto un determinato valore.

Questo secondo metodo sfrutta la capacità dell'interfaccia video di generare interruzioni quando il contenuto del raster register è uguale a un valore di confronto.

Il valore di confronto dev'essere inserito nello stesso raster register. Ad esso infatti corrispondono due distinti registri, uno a sola lettura, contenente il valore del contatore di linea, e uno a sola scrittura, contenente il valore da confrontare con il contatore di linea per generare l'interruzione.

Oltre alla preparazione del raster register, bisogna abilitare l'interruzione raster impostando a %1 il bit 0 del registro di abilitazione interruzione, in modo che possa essere generato un segnale d'interruzione quando il contatore di linea eguaglia il valore desiderato. Questo particolare argomento (le interruzioni del VIC) verrà ripreso in dettaglio fra breve.

Il programma che segue, utilizzando una procedura d'interruzione, determina un cambiamento nel colore del bordo quando il pennello elettroni

raggiunge circa la metà dello schermo.

Il programma si articola in due segmenti di codice: il primo abilita la procedura d'interruzione, disabilitando quella normale generata dal timer del CIA1.

Il secondo segmento di codice costituisce invece il corpo della procedura d'interruzione.

```
             .ORG $1000

Int.Vect     .EQU $0314       ;vettore d'interruzione
Cont.linea   .EQU $D012       ;raster register
Int.VIC      .EQU $D01A       ;abilitazione
                              ;interruzione VIC
Int.CIA      .EQU $DC0D       ;abilitazione
                              ;interruzione CIA 1
Rit.da.Int   .EQU $EA81       ;ritorno da interruzione

             SEI              ;maschera le interruzioni
                              ;nel microprocessore
Ripeti       LDA Cont.linea   ;attesa azzeramento
                              ;contatore
             BNE Ripeti       ;di linea

             LDA #$7F         ;disabilita le normali
             STA Int.CIA      ;interruzioni di CIA1

             LDA #<Inter      ;modifica il vettore
             STA Int.Vect     ;d'interruzione
                              ;affinche' punti
             LDA #>Inter      ;alla nuova routine
             STA Int+Vect+1

             LDA Val.linea    ;inserisce il nuovo
             STA Cont.linea   ;valore nel contatore
             LDA Cont.linea-1 ;di linea affinche' crei
                              ;un'interruzione quando
             AND #$7F         ;il conteggio raggiungera'
             STA Cont.linea-1 ;tale valore

             LDA #$01         ;abilita le interruzioni
                              ;sul VIC
             STA Int.VIC+1

             CLI              ;riabilita le interruzioni
                              ;sul microprocessore
             RTS              ;e ritorna

Inter        LDA Int.VIC      ;riabbassa IRQ riscrivendo
             STA Int.VIC      ;il registro
             INC Col.bordo    ;d'interruzione, modifica
             LDA Val.linea    ;il colore del bordo e
                              ;imposta il contatore di
             CLC              ;linea per la prossima
             ADC #$16         ;interruzione tra 16
                              ;linee di scansione
             STA Val.linea
             STA Cont.linea
             JMP Rit.da.Int   ;ritorna
                              ;dall'interruzione
Val.linea    .BYTE $00        ;valore del contatore di
                              ;linea a cui e' impostata
                              ;l'interruzione

             .END
```

Scroll fine dello schermo

L'interfaccia video consente anche lo scroll fine, ovvero la traslazione pixel per pixel del contenuto dello schermo sia in senso verticale sia in senso orizzontale.

Lo scroll orizzontale è controllato dai tre bit meno significativi del registro ubicato all'indirizzo $D016. Questi tre bit contengono un numero compreso fra 0 e 7 che indica il numero di pixel di cui il quadro è spostato verso destra.

Analogamente, il valore di scroll verticale è contenuto nei tre bit meno significativi di un secondo registro presente all'indirizzo $D011.

Oltre a questi bit, ve ne sono altri due che partecipano alla generazione dello scroll. Il bit 3 del registro $D016 controlla il numero di colonne visualizzate, che può essere 40 (bit impostato a %1) o 38 (bit impostato a %0), mentre il bit 3 nel registro $D011 controlla il numero delle righe visualizzate, che può essere 25 (bit impostato a %1), o 24 (bit impostato a %0).

Illustriamo ora la prassi generale per eseguire lo scroll orizzontale, da sinistra verso destra, lasciando al lettore ormai esperto la responsabilità di scrivere la necessaria routine.

Innanzitutto occorre diminuire il numero delle colonne visualizzate, impostando a %0 il bit 3 nel registro $D016. Così facendo, la prima colonna a sinistra e l'ultima colonna a destra, sempre presenti nella memoria di schermo, nella memoria colore e nella mappa di bit, vengono nascoste dal bordo.

In seguito, il registro di scroll orizzontale, (i tre bit meno significativi del registro $D016) dev'essere incrementato di un'unità per sette volte, traslando così tutta l'immagine verso destra di sette pixel. A questo punto si azzera nuovamente il registro di scroll orizzontale e si muove l'intero schermo (memoria di schermo o della mappa di bit, e memoria colore) di un carattere verso destra, tramite un'apposita routine, iniziando nuovamente la procedura di scroll.

Ognuno degli incrementi suddetti dev'essere eseguito in sincronismo con il contatore di linea, e preferibilmente con una procedura d'interruzione, in modo che la traslazione avvenga nel tempo che intercorre fra la rappresentazione di un quadro e la successiva.

Lo scroll verticale si ottiene in un modo del tutto analogo, utilizzando il registro $D011.

Il controllo delle interruzioni

Il meccanismo delle interruzioni consente all'interfaccia video d'interrompere il normale flusso di un programma chiamando una particolare routine di gestione dell'interruzione. In genere questa interruzione viene inoltrata quando si verifica una particolare condizione. Nel VIC sono presenti due registri che controllano l'abilitazione degli interrupt e la gestione delle procedure d'interruzione.

Il primo registro, di nome Enable Interrupt (abilitazione di interruzione), viene utilizzato per selezionare gli eventi esterni che possono originare interruzioni ed è allocato all'indirizzo $D01A. Per abilitare un interrupt è sufficiente impostare a %1 il corrispondente bit: quando si verifica la condizione impostata, il VIC genera un interrupt. Vediamo il significato di ogni singolo bit.

Il bit 0 del registro Enable Interrupt abilita le interruzioni originate quando il raster (contatore di linea) eguaglia il valore impostato.

Il bit 1 abilita le interruzioni originate dalle collisioni tra animazioni e caratteri.

Il bit 2 abilita le interruzioni generate dalle collisioni fra animazioni.

Il bit 3 infine abilita le interruzioni generate quando il pennello elettronico, ovvero il puntino generato sullo schermo durante la scansione dell'immagine, raggiunge una posizione "vista" dalla penna ottica.

Un secondo registro, l'interrupt register, ha il compito d'indicare alla routine di interrupt quale specifico evento ha generato l'interruzione, ed è allocato all'indirizzo $D019. A ogni bit è associata una possibile causa d'interruzione.

1) Il bit 0 dell'Interrupt Register viene impostato a %1 se l'interruzione è prodotta dal raster.

2) Il bit 1 viene impostato a %1 se l'interruzione è stata determinata da una collisione fra animazioni e caratteri.

3) Il bit 2 viene impostato a %1 per indicare che l'interruzione è stata determinata da una collisione fra animazioni.

4) Il bit 3 viene impostato a %1 per indicare che l'interruzione è stata determinata dalla penna ottica.

5) Infine il bit 4 viene semplicemente impostato a %1 in corrispondenza di una qualsiasi interruzione, a indicare che tale interruzione è stata originata dal VIC.

Quest'ultimo bit consente di verificare se l'interruzione è stata prodotta dal VIC, senza dover controllare ogni bit dell'interrupt register.

La routine di interrupt può accedere all'interrupt register per determinare la causa dell'interruzione e chiamare le appropriate routine. Quando questa routine accerta un interrupt, deve azzerare il corrispondente bit nell'interrupt register in modo che possa nuovamente tornare a %1 alla successiva interruzione. ■

ARRETRATI

COMPLETATE LA VOSTRA COLLEZIONE

NUMERO 1/86 - Telecomunicazioni: guida all'acquisto di un modem. Il disk drive 1541 ed i suoi limiti tecnici. Corso di programmazione in L.M. su C-64. Il C-128: prova. Il nuovo Commodore Amiga a confronto con IBM PC, AT e Macintosh. Reset per il 1541. Amiga, dove fantasia e realtà si incontrano. Fare musica con il C-128. Convertitore grafico per Commodore 64: listato. Cultura e informatica. Software Gallery: Lode Runner Rescue, The Hitchhiker's Guide to the Galaxy, Jet, Scenery Disks. Software Helpline: The Hitchhiker's Guide to the Galaxy, Zaxxon, Wolfenstein, Star Wars, Ghostbusters, Jumpman, Mindshadow e Tracer Sanction.

NUMERO 2/86 - Jack Tramiel. Corso di programmazione in L.M. su C-64. Telecomunicazioni. Il C-128D: Come operare all'interno del disk drive. Computer e grafica. La Commodore e la didattica. La Rom del C-64. Amiga e Atari 520ST a confronto. Speciale USA: il CES di Las Vegas; il mondo del III Commodore Show di San Francisco. La mappa di memoria del C-128. Fare musica con il 64: Rondò Veneziano. Archivio programmi: listato per C-64. Software Gallery: Turbo Loading Cartridge, Machine Lightning, Basic Lightning, Uridium. Software Helpline: The Hitchhiker's Guide to the Galaxy, Beyond Castle Wolfenstein, Ghostbusters, Cosmic Balance, Jumpman, Star Trek, Avventura nel computer.

NUMERO 3/86 - Nuovi prodotti Commodore: 64C, 1581 e 1802. Linea Commodore PC IBM compatibili. Gli Ampersand file. Geos. Corso di programmazione in L.M. su C-64. Il SIM HI-FI IVES. Software in CP/M per C-128. Amiga: una nuova era nel mondo del PC. La A-Squared e AmigaLive! Le piante parlano: progetto hardware/software. Amiga: tra sogno e realtà. Espansioni di memoria per il C-128. I disk drive 1570 e 1571 a confronto. Prom ed Eprom. Il suono nel C-128. Jenny: listato per C-128. Eseguiamo le routine del 1541. Suoni e musica con l'AmigaBASIC. Software Gallery: Di-Sector V3.0, Project: Space Station, Matrix 128, 3D Graphics Drawing Board, Textcraft, Game Killer, Nexus, Mach 128, 1571 Clone Machine, VizaStar e VizaWrite, The Final Cartridge. Software Helpline: The Dallas Quest, Questron.

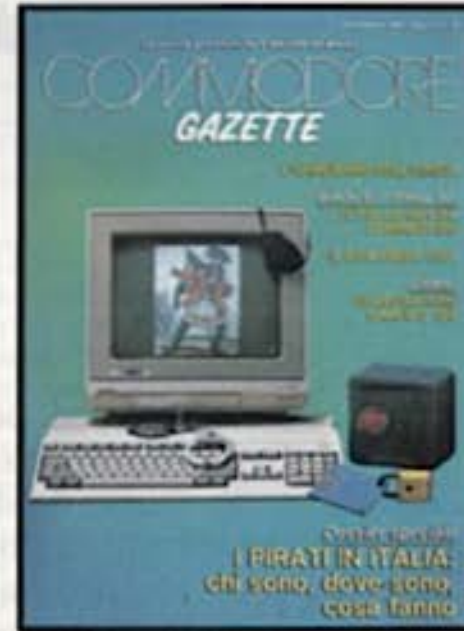
NUMERO 4/86 - Smau '86. CLI: Command Line Interface. La compatibilità IBM per Amiga: Sidecar e Transformer. Schema interno dell'Amiga. Genlock: effetti speciali audio e video con l'Amiga. Amiga days '86. Novità Commodore. Geos e il suo creatore: intervista in esclusiva con il presidente della Berkeley Softworks. Personalizziamo il sistema operativo del C-64. Csa turbo Amiga. La memoria di massa e i compact disk. Terzo Commodore Show, Los Angeles. Tempo di Biennale, tempo di Amiga. Protezione scrittura e interruttore per il numero di device: progetto hardware per C-64. Software per Amiga: descrizione di più di 250 programmi. Grafica 128: listato per C-128. Corso di programmazione in L.M. su C-64. Lista delle routine Kernel del C-128. Software Gallery: Johnny Reb II, War Play, Time Trax, Aegis Impact, The Cataloger, Cartuccia ES-9, Online!, Deluxe Paint. Software Helpline: Borrowed Time.

NUMERO 1/87 - Il Commodore 64 a Hollywood: l'impiego del 64 nel mondo dello spettacolo. L'Amiga nella realizzazione del serial «Amazing Stories». Amiga e Pontaccio: un Tandem vincente. La politica economica della Commodore International: una panoramica sulle condizioni finanziarie della CBM. Gli Idea Processor: utility per l'organizzazione dei propri pensieri. World of Commodore: servizio speciale sulla IV edizione della nota rassegna canadese. Amiga 1060 Sidecar. La grafica e l'Amiga: un'introduzione alle istruzioni grafiche dell'AmigaBASIC. Grafica avanzata per il C-128: programmazione del chip grafico 8563. Geos: le chiavi del regno. Super Basket NBA: simulazione strategica del gioco del basket per C-64/128. Commodore PC 40 AT: un prodotto IBM compatibile all'insegna della tecnologia avanzata. Hardcopy della pagina grafica in alta risoluzione 640x200 pixel per C-128. Gestione magazzino per Commodore 128. Novità e nuovi prodotti. Routine alternate Kernel-Disco. La stampante MPS 1000. Corso di programmazione in L.M. su C-64. Software Gallery: Tomahawk, Flight Simulator II, Partner 128, Deluxe Paint II, Defender of the crown, Blood'n guts, '43 one year after, Mind Walker, Thai boxing, Skyfox. Software Helpline: Flight Simulator II, View to a kill, Ultima IV.

NUMERO 2/87 - Il Consumer Electronics Show: servizio speciale sulla nota rassegna di Las Vegas. Nuovi nati nella famiglia Commodore: Amiga 500 e 2000. Nuove istruzioni del chip 8502 utilizzato come CPU nel Commodore 128. La videodigitalizzazione: un'analisi del processo di video-digitalizzazione unito all'esame di due noti digitalizzatori per C-64/128 e Amiga. Batch file con l'Amiga: introduzione alle potenzialità operative dell'AmigaDOS. Corso di programmazione: impariamo a programmare in linguaggio macchina il Commodore 64. La produzione di «The Bard's Tale»: intervista in esclusiva per Commodore Gazette. Gestione di un club: listato per C-64. Protector 128: protettore di programmi per C-128. Geos: le chiavi del regno di questo nuovo sistema operativo. Nuove potenzialità per le macchine a 8 bit della Commodore. Copie bit a bit sul 1541. Software Gallery: American Challenge e America's Cup, Page Setter, The Inheritance, Webstars, Vera Cruz, [illegible] Orange, S.D.I., Sinbad, Turbo Pascal, EDNA, Power Cartridge. Software [illegible] Oo-Topos.

NUMERO 3/87 - Commodore Italiana: gli assi nella manica per il 1987. I nuovi Commodore Amiga 500 e 2000: alla scoperta delle potenzialità operative dei nuovi nati in casa Commodore. CeBit '87: servizio speciale sui nuovi prodotti presentati dalla Commodore Tedesca. Roma Ufficio '87. La famiglia di programmi Geos: le applicazioni dedicate e la nuova versione per C-128. Corso di Programmazione in L.M. su C-64. Find 128: utility per C-128. Commodore Show San Francisco: servizio speciale sulla nota rassegna californiana. Il Software Sistema dell'Amiga. Catalogazione Dischi per C-128: listato per C-128. Il disk drive OC-118: prova hardware. Dump 128: listato. Velocizzatori e disk drive 1541. Scroll list 128: listato. Software Gallery: Balance of Power, Silent Service, Portal, Graphic Adventure Creator, Melody Hall's Printware Series, Cyborg, Twin Tornado, Thanatos, The Ket Trilogy. Software Helpline: Leather Goddesses of Phobos.

NUMERO 4/87 - Obiettivo Telematica: la telematica spalanca le sue porte al C-64 con il Videotel e le PGE. La Commodore cambia volto. Basic 8.0: sistema grafico per Commodore 128 che consente risoluzioni di 640x200 pixel e fornisce 50 nuovi comandi Basic. Desktop publishing con l'Amiga: rassegna di programmi di desktop publishing realizzati per l'Amiga. La stampante MPS 1200: prova hardware della nuova periferica Commodore. Amigraf: listato in Amiga BASIC per tracciare il grafico di funzioni matematiche. Il Consumer Electronics Show di Chicago: Servizio speciale. Nuovi comandi e Ram Disk: Utility per C-128. Disk Sector Editor: listato per C-128. L'hardware dell'Amiga. Bank Data: archivio di nomi e indirizzi per C-128. Corso di Programmazione in LM per C-64. Software Gallery: Bureaucracy, Faery Tale, Uninvited, Murder on the Atlantic, Deathscape, Frankenstein, Up Periscope!, Barbarian, 500 cc Grand Prix, Logistix, Superbase. Software Helpline: A View to a Kill.

NUMERO 5/87 - Computer grafica nella ricerca: l'uso dell'Amiga nel campo della ricerca scientifica. Software Musicale per l'Amiga. L'Amiga 2000 e la compatibilità IBM XT attraverso la scheda A2088. Inventando l'Amiga: il travagliato avvio commerciale del gioiello della Commodore. Corso di programmazione in L.M. per C-64. L'Hardware dell'Amiga: alla scoperta delle caratteristiche strutturali dell'Amiga. Il Comdex di Atlanta: servizio speciale. Codici in data C-128: utility per Commodore 128. Amiga 3D: animazione grafica tridimensionale. Append Merge: listato per C-128. Semper Sperum: listato per C-64 per la pianificazione delle giocate al Lotto. Bank Data. La produzione di Adventure nel nostro Paese: intervista con la Arscom. Software Gallery: the Mirror Hacker package, ProWrite, Guild of Thieves, Barbarian, Defender of the Crown, Denarius, Revs Plus, Inheritance 2, Kinetik, The Three Musketeers, Galileo. Software Helpline: Trinity.

NUMERO 6/87 - SIM Hi-Fi Ives '87. SMAU '87: le novità di casa Commodore. Dossier speciale: i pirati in Italia. La vera storia del gruppo 2703, un nuovo business: l'hardware pirata, si io sono un pirata pentito, Lago: la riscossa degli importatori, contro i pirati prezzi più bassi, Niwa: quando la copia costa più dell'originale, Assoft: il periodo dell'impunità è finito. Prove hardware: il disk drive 1581. Telematica: un magazine in Videotel per il C-64: Lasernet. Amiga software: le nuove frontiere dei giochi d'avventura. Toolkit 128: listato per i nuovi comandi BASIC. Corso di programmazione in LM per C-64: il sistema BCD. Amiga hardware: gli sprite e il blitter. I nuovi prodotti USA per Amiga. Elab per C-128: listato di un programma grafico. Software Gallery: ACE 2, Pirates!, Passengers on the wind, The lurking horror, Hollywood hi-jinx, Vampyr Motel, I predatori della Valle dei Re, Vizawrite desktop, 007: The living daylights, Solomon's key. Software Helpline: The lurking horror, Bureaucracy.

NUMERO 7/87 - Le rosee promesse del 1988: intervista ai manager della Commodore. I nuovi prodotti per Amiga dei saloni USA: AmiEXPO, Commodore Show e Comdex 1987. Un C-64 dentro il vostro Amiga: anteprima sugli emulatori di C-64. L'Aids del computer: i programmi virus. Fred Fish: i programmi di Pubblico Dominio per Amiga. Alla scoperta della geometria frattale: le teorie di Mandelbrot illustrate e applicate da un programma in Amiga Basic. Le espansioni di memoria 1764, 1700 e 1750 per C-64 e 128. Prova su strada del Commodore PC1. Videoscape 3-D: comincia l'era del desktop video. C-128: l'avvenire si chiama Geos: il S.O. della Berkeley Softworks valutato in versione 128. Amiga hardware: ultima puntata. Corso di programmazione in LM per C-64: assemblatori e codice sorgente. Software Gallery: the Advanced OCP art studio, Alien fires, Echelon, Bubble bobble, Knight orc, Quedex, Fire power, Test drive, Video title shop, The hunt for Red October. Software Helpline: The three musketeers.

NUMERO 1/88 - Novità dagli Usa per l'Amiga: AmiEXPO e World of Commodore Show. Intervista con Dale Luck sui nuovi Kickstart e Workbench 1.3. Speciale telematica: il 6499 alla prova, adattatore telematico 6499 domande e risposte, Amiga e Videotel, Hacking tra tecnica e arte, intervista con Hugo Cornwall, glossario: la telematica in 60 parole chiave. Wb e .info file: a cosa servono e come usufruirne. Il drago e il computer: un viaggio nell'universo dei giochi di ruolo. Videogiochi e apprendimento: il quinto convegno Automat. Geos: anatomia di un sistema operativo. Corso di programmazione in LM per C64: le interfacce col mondo esterno. Equo canone C-64: listato. Osserviamo gli astri attraverso il C-128: listato. Software Gallery: Airbone ranger, Apollo 18: mission to the moon, Chuck Yeager's advanced flight trainer, Auto duel, Moebius, Shoot'em up construction kit, The art of chess, World tour golf, Bocce, Tombola, Flying shark, Mini putt, Crazy car, Test drive. Software Helpline: Neverending story, Stationfall.

NUMERO 2/88 - Intervista con Dan Silva, l'autore di Deluxe Paint. Speciale simulazione del volo: Flight Simulator II, il funzionamento di Flight Simulator II, Co-Pilot un libro da volare, il volo a schema, simulatori di volo a confronto. Reportage dal CEBIT 88, la più grande fiera europea dell'informatica. Sconfitta l'AIDS dell'Amiga, con il listato del programma antivirus "Guardian". Prova hardware: hard disk per l'Amiga. geoPublish, il desktop publisher GEOS compatibile. Corso di programmazione in LM per il C-64. Pangea Basic C-128, listato. Super Quark C-128, listato. Il progetto ARP. Software Gallery: Shadowgate, Stratton, Yogi Bear, Terramex, Ports of Call, The Graphics Studio, Jet, John Brenner Boston Celtic, Power at Sea, Stealth Fighter, Card Sharks, Mercenary, Strike Fleet, King of Chicago, Jinxter. Software Helpline: Perry Mason and the Case of the Mandarin Murderer.

NOME E COGNOME ....

INDIRIZZO ....

CITTÀ .... **C.A.P** ....

DESIDERO RICEVERE I SEGUENTI NUMERI

ARRETRATI: ....

PER UN TOTALE DI LIRE (lit. 14.000 ognuno): ....

☐ **ALLEGO ASSEGNO BANCARIO O CIRCOLARE O POSTALE**

☐ **ALLEGO FOTOCOPIA DELLA RICEVUTA DEL VAGLIA POSTALE**

FIRMA ....

RITAGLIARE (SI ACCETTANO ANCHE FOTOCOPIE) E SPEDIRE A:

COMMODORE GAZETTE - Via Monte Napoleone, 9 20121 Mi

FILO DIRETTO CON LA CBM

LA COMMODORE RISPONDE AI SUOI UTENTI

VALUTARE L'USATO

Sono un futuro acquirente di Amiga 500 (o anche 2000). Vorrei sapere se la Casa Madre, cioè la Commodore, è d'accordo nel riprendere indietro il vecchio hardware valutandolo, e con i soldi ricevuti dalla valutazione poter comprare uno dei nuovi straordinari Amiga, che comprerei via posta direttamente dalla fabbrica di Milano.

Fabio Testaferrata
Vaiano (FI)

Per qualsiasi informazione sul ritiro dell'usato sia Commodore che non Commodore, è possibile chiamare direttamente l'ufficio HOTLINE, disponibile per qualsiasi chiarimento sulla campagna promozionale, tutti i giorni dalle 9.00 alle 13.00 e dalle 14.00 alle 18.00. (Tel. 02/ 61832245-61832246-618321)

DOMANDE A RAFFICA

1) Utilizzando il *Workbench 1.2* come si ottiene il segno "^"? A me sembra importante, soprattutto per chi programma in Basic.

2) Sempre utilizzando lo stesso *Workbench*, si può passare dalla bassa risoluzione a quella alta?

3) Cosa dobbiamo fare noi poveri utenti che vogliamo scriverci un programmino in Basic quando l'Editor segnale "Syntax Error" e il famigerato quadrato arancione racchiude un'area completamente vuota?

4) La Vostra pubblicità segnala che all'interno del mio Amiga 2000 esiste una batteria tampone in grado di alimentare l'orologio interno. Francamente la mia batteria (nonostante un mese di fermo per riparazione presso un Vostro laboratorio) non ne vuol sapere di funzionare per più di tre ore. Cosa ci posso fare?

Mauro Ghiglia
Carcare (SV)

1) Per ottenere il simbolo "^" con la tastiera italiana, si devono premere prima i tasti SHIFT e "ì", poi la barra spaziatrice.

2) Per poter ottenere il Workbench in interlace 640 x 512, si deve caricare il programma Preferences presente sul disco e mettere in ON l'opzione "Workbench Interlace". Poi si deve salvare la configurazione con il comando SAVE e far ripartire il sistema. Quando si ricarica il Workbench ci si trova in interlace.

3) Quando il Basic segnala un errore del tipo sopra descritto, ci si deve assicurare che su quella linea non vi siano caratteri non riconosciuti dal Basic. Se l'errore si dovesse ripresentare, per individuarlo, utilizzare il comando TRON e abilitare la finestra list durante il RUN.

4) Se l'orologio interno non funziona, le consigliamo di ritornare al Centro Assistenza (assicurarsi che sia autorizzato Commodore) e, dopo aver spiegato bene il problema, farlo riparare.

Questa è una rubrica indipendente gestita dalla Commodore Italiana. Le risposte ai quesiti dei lettori sono redatte dai responsabili della CBM e rappresentano esclusivamente le opinioni di detta società, le quali possono anche discostarsi da quelle della redazione di Commodore Gazette.

Indirizzate tutta la corrispondenza per questa rubrica a:

COMODORE GAZETTE
Filo diretto
con la Commodore
Via Monte Napoleone, 9
20121 Milano

I PREZZI DI LISTINO DEI PRODOTTI COMMODORE

PRODOTTO		PREZZO AL PUBBLICO IVA ESCLUSA
C-64	HOME COMPUTER 8 bit, 64K RAM. Collegabile a un qualsiasi televisore	**L. 325.000**
C-128D	HOME COMPUTER 8 bit, 128K RAM. Display 40 o 80 colonne (RGB). Programmabile in CP/M 3.0. Funzionante anche in modo C-64. Disk drive 1571 da 340K incorporato	**L. 895.000**
A500	HOME COMPUTER 16/32 bit, 512K RAM. Comprende un disk drive da 3,5" da 880K e un mouse	**L. 950.000**
A2000	PERSONAL COMPUTER 16/32 bit, 1 MB RAM. Il sistema può essere espanso tramite schede da inserirsi in appositi slot interni e reso anche IBM XT e AT compatibile. Possibilità di un secondo drive interno da 3,5" o hard disk e di un drive interno da 5,25". Comprende il mouse	**L. 2.005.000**
PC 1	PERSONAL COMPUTER. Microprocessore Intel 8088 a 16 bit, memoria RAM da 512K sistema operativo MS-DOS 3.2. Scheda video con emulazioni MDA/CGA/Hercules. Un disk drive da 5,25", monitor a fosfori verdi da 12". Porta seriale RS232C e parallela	**L. 945.000**
PC 10-III	PERSONAL COMPUTER. Microprocessore Intel 8088 a 16 bit; coprocessore matematico Intel 8087 opzionale; memoria RAM da 640K; memoria ROM (BIOS) da 8K; frequenza di clock selezionabile 4,77 MHz, 7,16 MHz e 9,54 MHz. Sistema operativo MS-DOS 3.2. Scheda video monocromatico/colore con emulazioni MDA/CGA/Hercules. Due floppy disk drive da 360K. Monitor monocromatico a fosfori verdi da 12". Porta seriale RS-232C e parallela Centronics. Tastiera avanzata e orologio-calendario a tampone	**L. 1.490.000**
PC 10-III C	Stessa configurazione ma con monitor 14" a colori mod. 1084	**L. 1.890.000**
PC 20-III	PERSONAL COMPUTER. Microprocessore Intel 8088 a 16 bit; coprocessore matematico Intel 8087 opzionale; memoria RAM da 640K; memoria ROM (BIOS) da 8K. Frequenza di clock selezionabile 4,77 MHz, 7,16 MHz e 9,54 MHz. Sistema operativo MS-DOS 3.2. Scheda video monocromatico/colore con emulazioni MDA/CGA/Hercules. Un floppy disk drive da 360K e un hard disk da 20 MB standard. Monitor monocromatico a fosfori verdi da 12". Porta seriale RS-232C e parallela Centronics. Tastiera avanzata e orologio-calendario a tampone	**L. 2.240.000**
PC 20 -III C	Stessa configurazione ma con monitor 14" a colori mod. 1084	**L. 2.640.000**
PC 40/20 AT	PERSONAL COMPUTER. Microprocessore 80286 a 16 bit; clock 10/6 MHz; memoria RAM da 1 MB; Hard disk da 20 MB; un disk drive da 1,2 MB; sistema operativo MS-DOS 3.2; scheda video monocromatico/colore 132 colonne AGA di serie; monitor monocromatico a fosfori verdi da 14"; porta seriale RS-232C e parallela Centronics	**L. 3.490.000**
PC 40/40 AT	PERSONAL COMPUTER. Microprocessore Intel 80286 a 16 bit; clock 10/6 MHz; memoria RAM da 1 MB. Un floppy disk drive da 1,2 MB e hard disk da 40 MB. Sistema operativo MS-DOS 3.2. Scheda video monocromatico/colore 132 colonne AGA di serie. Monitor monocromatico a fosfori verdi da 14". Porta seriale RS-232C e parallela Centronics	**L. 4.490.000**
PC 60/40	PERSONAL COMPUTER. Microprocessore Intel 80386 a 32 bit; coprocessore matematico Intel 80387 opzionale; memoria RAM da 2,5 MB. Un disk drive da 1,2 MB e un hard disk da 40 MB. Sistema operativo MS-DOS 3.2, MS-DOS windows 386 opzionale. Scheda grafica di tipo EGA standard di serie. Monitor monocromatico di tipo ADI a fosfori verdi da 14". Due porte seriali RS-232C e due parallele Centronics	**L. 8.490.000**
PC 60/80	Stessa configurazione del PC 60/40 ma con hard disk da 80 MB e, in più, MS-DOS WINDOWS 396 e MOUSE	**L. 9.990.000**
	DISK DRIVE	
1541 II	DISK DRIVE SLIM da 5,25" (170K) dedicato al C-64/128	**L. 395.000**
1571	DISK DRIVE da 5,25" (340K o 410K sotto CP/M) dedicato al C-128	**L. 460.000**
1581	DISK DRIVE da 3,5" (800K) dedicato al C-64 e al C-128	**L. 420.000**
K 910	DISK DRIVE da 3,5" (1,44 MB) dedicato al PC 10/20-III	**L. 295.000**
A1010	DISK DRIVE da 3,5" dedicato all'Amiga 500/1000/2000	**L. 390.000**
A 2010	DISK DRIVE INTERNO da 3,5" dedicato all'A2000	**L. 290.000**
	MONITOR	
1084	MONITOR A COLORI alta risoluzione, 14", con audio, antiriflesso. Dedicato al C-64, al C-128 e all'Amiga 500/1000/2000	**L. 545.000**
1402	MONITOR monocromatico da 12"	**L. 255.000**

1802	MONITOR A COLORI alta risoluzione, 14", con audio. Dedicato al C-64 e al C-128 (40 colonne)	**L. 445.000**
2080	MONITOR A COLORI AD ALTA PERSISTENZA alta risoluzione, 14", con audio, antiriflesso. Dedicato al C-64, C-128, PC e all'Amiga 500/1000/2000	**L. 630.000**
	STAMPANTI	
MPS 1200S	STAMPANTE 80 colonne, 120 cps, bidirezionale, carta in modulo singolo e trascinamento modulo continuo. Dedicata al C-64 e al C-128	**L. 480.000**
MPS 1200P	STAMPANTE a 9 aghi 120 cps; bidirezionale; 80 colonne; carta in modulo singolo e trascinamento modulo continuo. Near letter quality. Interfaccia Centronics per PC 10/20 II - PC 40 AT e Amiga 500/1000/2000	**L. 480.000**
MPS 1250	STAMPANTE stesse caratteristiche della MPS 1200P ma con interfacce seriale RS-232 C e parallela Centronics di serie	**L. 495.000**
MPS 1500	STAMPANTE A COLORI 80 colonne, 130 cps, 4 colori, bidirezionale, carta in modulo singolo e trascinamento modulo continuo	**L. 550.000**
	ACCESSORI	
PC EXP1	BOX esterno per espansione PC 1 può contenere 3 schede aggiuntive e hard disk; alimentazione e cavi compresi	**L. 590.000**
1530	REGISTRATORE dedicato al C-64 e C-128	**L. 55.000**
1764	ESPANSIONE DI MEMORIA per il C-64 da 256K	**L. 195.000**
1700	ESPANSIONE DI MEMORIA per il C-128 da 128K	**L. 165.000**
1750	ESPANSIONE DI MEMORIA per il C-128 da 512K	**L. 235.000**
1311	JOYSTICK	**L. 13.500**
1399	JOYSTICK 8 microswitch auto-fire	**L. 29.000**
1351	MOUSE dedicato al C-64 e al C-128	**L. 99.000**
1352	MOUSE due tasti collegabile alla serie dei PC; include MICROSOFT MOUSE DRIVER	**L. 72.000**
6499	ADATTATORE TELEMATICO OMOLOGATO per C-64. Consente il collegamento al Videotel, P.G.E. e alle banche dati	**L. 149.000**
A 501	ESPANSIONE DI MEMORIA PER L'AMIGA 500 da 512K e orologio	**L. 212.500**
A 520/1	MODULATORE VIDEO PER L'AMIGA 500. Permette di collegare l'A500 a un qualsiasi televisore b/n o colore	**L. 40.000**
	Cavo collegamento Amiga 500 con TV presscart	**L. 27.000**
A2052	SCHEDA DI ESPANSIONE DI MEMORIA da 2 MB per l'A2000	**L. 590.000**
A2060	Scheda video/RF Modulatore per TV e VCR per l'A2000	**L. 138.000**
A2092	Hard disk da 20 MB settorizzabile PC/Amiga + scheda controller hard disk	**L. 845.000**
A2088+A2020	Scheda Janus XT compatibile + drive interno da 5,25"	**L. 895.000**
A2286+A2020	Scheda Janus II AT compatibile + drive interno da 5,25"	**L. 1.285.000**
A2090+A2092	Hard disk 20 MB, interfaccia SCSI per A2000	**L. 1.025.000**
A2090+A2094	Hard disk 40 MB, interfaccia SCSI per A2000	**L. 1.575.000**
A2300	GENLOCK card semiprofessionale per A2000; permette di miscelare grafici con videocamere VCR; selettore per video Amiga, monitor esterni, proiettori; include software per titolazione	**L. 345.000**
A2350	Professional Video Adapter Card per A2000. Genlock professionale con freeze frame e digitalizzazione video; include software di controllo per la gestione interattiva	**L. 1.645.000**

COMMODORE NEWS

NOVITÀ HARD E SOFTWARE DALL'ITALIA E DAL MONDO

ITALIA

CAMBIAMENTI AL VERTICE

L'ingegner Sergio Simonelli è il nuovo Amministratore Delegato della Commodore Italiana.

Nato ad Arcola (SP) nel '49, laureato a Genova in ingegneria elettronica, vanta significative esperienze nel settore informatico: dal 1978 al 1981 opera presso la Direzione vendite della Data General, per i successivi quattro anni in Perkin Elmer, Divisione calcolatori, prima come Direttore vendite, poi come Direttore di divisione; il biennio successivo lo vede come Amministratore Delegato e Direttore Generale della Prime Computer.

L'inserimento dell'ingegner Simonelli al vertice della Commodore Italiana avviene in un momento strategicamente importante per la Società che, da leader assoluta nel campo degli home computer in Italia e nel mondo, si è inserita prepotentemente anche nel settore dei sistemi professionali.

In questo nuovo campo è ora impegnata a consolidare una presenza già significativa.

ACTIVISION E LEADER

Con decorrenza dal 10 aprile 1988, a seguito di accordi presi fra Rod Cousens, vicepresidente dell'Activision Europa, e John Holder, amministratore della Leader Distribuzione, è stato siglato un contratto di esclusiva per la distribuzione in Italia dei prodotti Activision.

In relazione a tale contratto, la Leader si impegna a promuovere i prodotti Activision con un adeguato supporto promozionale e pubblicitario che sicuramente creerà le premesse per un sostanziale aumento del volume delle vendite. Da parte sua, l'Activision ha in programma per i prossimi 12 mesi l'uscita di 35 titoli nuovi, tra cui il best-seller *After Burner* (su licenza Sega), che già da ora viene considerato come il numero uno del prossimo Natale.

Leader Distribuzione
Via Mazzini, 15
21020 Casciago (VA)
(tel. 0332/212255)

GEOS IN ITALIANO

Il sistema operativo *GEOS*, di cui la IHT Gruppo Editoriale pubblica la *Guida ufficiale alla programmazione* (in libreria da luglio), è ora disponibile anche con una serie di manuali in italiano.

Esistono già tre pacchetti, che illustriamo brevemente.

GEOS: la confezione contiene la versione 1.3 del sistema operativo; *geoPaint* (un completo laboratorio grafico a colori); *geoWrite V1.3* (un elaboratore di testo dotato di 5 caratteri tipografici e diversi corpi, utilizzabile anche con i disegni creati da *geoPaint*); una serie di accessori da scrivania come orologio, calcolatrice, archivio per immagini e testi.

GEOWRITE WORKSHOP: oltre alla più aggiornata versione 2.1 di *geoWrite* contiene altre utility come *geoMerge* (versatile programma di integrazione della corrispondenza), *geoLaser* (per utilizzare la stampante Apple LaserWriter), *Text Grabber* (per importare e convertire documento preparati da altri elaboratori di testi).

DESKPACK: una raccolta di applicazioni e utility ampiamente descritta in questo stesso numero di *Commodore Gazette*.

Leader Distribuzione
Via Mazzini, 15
21020 Casciago (VA)
(tel. 0332/212255)

COMMODORE NEWS
NOVITÀ HARD E SOFTWARE DALL'ITALIA E DAL MONDO

COMUNICATO COMMODORE

La Commodore avvisa che sono disponibili i manuali in italiano dell'Amiga 500 e di tutti gli altri prodotti. Chi avesse il manuale in inglese può ricevere la versione italiana gratuitamente. Unica condizione: inviare la garanzia o una fotocopia della stessa, allegando un biglietto di richiesta.

La Commodore Italiana ricorda che i prodotti con garanzia sono gli unici a poter usufruire del servizio assistenza.

Commodore Italiana S.p.A.
Via F.lli Gracchi, 48
20092 Cinisello Balsamo (MI)
(tel. 02/618321)

DAL LISTINO LEADER

Segnaliamo alcuni recenti programmi per il C-64 e per l'Amiga in distribuzione o annunciati dalla Leader.

Daley Thompson '88: otto discipline sportive dalla grafica sbalorditiva per un gioco dal sapore olimpico (Ocean).

Driller: il primo prodotto che utilizza un simulatore di ambiente solido tridimensionale (Incentive).

Operation Wolf: il coin-up dell'anno, grafica ed effetti incredibili (Ocean).

Virus: a bordo di un'astronave, attraverso paesaggi tridimensionali, il giocatore affronta alieni che minacciano i pianeti con terribili virus rossi (Firebird).

Yety: sulle montagne dell'Himalaya, alla ricerca della mitica creatura; un'avventura non certo facile perché lungo il cammino il giocatore dovrà scontrarsi con i discepoli del Dali Lama, che proteggono lo Yety (Destiny).

Robocop: al momento il lavoro dei programmatori è top secret, ma – a quanto si dice – la versione per computer non sarà inferiore al film (Ocean).

Typhoon: dà la possibilità di sfidare contemporaneamente 12 avversari; bombe, missili, laser, potenziano l'equipaggiamento per sconfiggere gli invasori (Ocean).

Mindfighter: basato sull'omonimo libro (che si trova all'interno della confezione), ricrea l'atmosfera originale della storia. Protagonisti sono quattro studenti di psicologia, uno dei quali – Robin – possiede un eccezionale controllo della sua mente. Un giorno Robin si sveglia in uno strano mondo sconosciuto e scopre di trovarsi nel futuro (Activision).

Vindicator: la naturale prosecuzione di *Green Beret*; tre livelli ambientati in un paesaggio segnato dalla guerra (Ocean).

Leader Distribuzione
Via Mazzini, 15
21020 Casciago (VA)
(tel. 0332/212255)

ESTERO

TV SHOW & TV TEXT

TV Show è un programma per la creazione di presentazioni animate su Commodore Amiga o tramite videoregistratore. Offre più di 50 effetti speciali applicabili su ogni soggetto provenienti da un qualsiasi formato grafico IFF (compreso quello HAM). Si dimostra molto flessibile: è compatibile con ogni membro della famiglia Amiga, e può funzionare altrettanto bene in formato PAL e in formato NTSC. Può essere anche usato con *Genlock* per produzioni video, realizzando schermate di più di 700 x 480 pixel.

TV Text offre invece le funzioni del generatore di titoli, utile per completare (tramite *Genlock*) una produzione video. Dà supporto al mouse, all'intera palette Amiga di 4096 colori, a una risoluzione video media, alta o overscan (a pieno schermo), alle immagini IFF e agli standard PAL e NTSC e può servirsi di qualsiasi carattere generato dall'Amiga.

I due pacchetti sono distribuiti in Italia dalla Lago.

Lago snc
Via Napoleona 16
22100 Como
(tel. 031/300214)

INFORA 88

A Lione, nella nuova città delle esposizioni "Eurexpo", si tiene ogni anno *Infora* il più importante salone francese d'informatica e uno dei più importanti d'Europa.

Quest'anno, parallelamente a Infora 88, ci sarà anche un salone intitolato "Mediatica" che abbraccerà tutti gli strumenti del marketing e la loro trasmissione. Ma il cavallo di battaglia di Infora rimane l'hardware: al suo pubblico di quadri superiori, dirigenti, tecnici, quadri intermedi e pubblici amministratori (infatti il salone non è aperto al pubblico generico) si presenterà suddivisa nelle sezioni: ***micro e mininformatica, software, telematica, burotica e organizzazione aziendale, periferiche e accessori, servizi bancari***. Le aziende interessate dovranno rivolgersi al seguente indirizzo.

Delegazione Italiana dell'Ente Fiere di Lione
Casella Postale 1536
16100 Genova
(tel. 010/386006; 050/532203)

ELECTRONIC ARTS NEWS

La Electronic Arts ha annunciato di aver ottenuto la distribuzione in esclusiva per l'Europa del software pubblicato dalla Strategic Studies Group, una casa produttrice australiana, leader nel mercato dei wargame e dei giochi di science fiction. I titoli disponibili verranno distribuiti in Europa a un prezzo più basso. In maggio è già stata realizzata l'edizione

primi nove titoli. Cinque hanno come sfondo la Seconda guerra mondiale: *Carriers of War* ne ricrea le battaglie navali, *Europe Ablaze* si sofferma sui panorami della guerra aerea tra Francia e Germania, *Battlefront* e il suo seguito *Battles in Normandy* ripercorrono le grandi battaglie, mentre *Russia* ricorda i quattro amari anni di conflitto nei Paesi dell'Est.

Tra gli altri titoli segnaliamo almeno *Reach to the Stars* che trasporta invece il giocatore nel futuro, e gli assegna il ruolo di unico responsabile della sopravvivenza della sua razza. Alla ricerca di nuovi pianeti da colonizzare, il protagonista avrà a disposizione tutte le risorse sociali, economiche e militari dell'umanità.

Tutti i titoli sono disponibili nella versione per il C-64 su disco ($ 18.95). Dell'ultimo esiste anche la versione per l'Amiga ($ 24.95).

Altro importante annuncio della Electronic Arts: è disponibile la versione PAL di *Deluxe Video 1.2*. Come molti già sanno, questo programma mette in grado di creare animazioni con l'Amiga, testi tridimensionali, effetti speciali e così via. *Deluxe Video* accetta disegni da *Deluxe Paint*, *Deluxe Print*, e da dischi che contengono suoni ed effetti acustici generati da *Deluxe Music Construction Set* e da *Instant Music*. Nel pacchetto attuale sono uniti altri dischi di postproduzione, contenenti nuove immagini, musiche ed effetti sonori utili per la produzione di video.

Electronic Arts Ltd.
11-49 Station Road, Langley
Berkshire SL3 8YN, England
(tel. 0753/46672) ■

ERRATA CORRIGE

Silvio Zuanetti, di Padova, ci segnala un'errata corrige per il listato ***Equo canone***, a **pagina 98** del numero **1/88**: la linea 5360 dev'essere modificata come segue:

```
5360 IF V=0 AND ((ASC(F$) <> 39 AND ASC(F$) <= 64) OR (ASC(F$) > 97)) THEN 5200
```

Le righe 7500, 7501, 7510 a **pagina 99** devono essere modificate come segue:

```
7500 IF KK <= 87 THEN RETURN
7501 PRINT"[illegible]INSERISCI IL COEFFICIENTE D'AUMENTO[illegible]DELL'ANNO 19";IP$
7510 PRINT"[illegible]COEFFICIENTE % ==>";:V=1:GOSUB 5270:PE=VAL(IP$):KK=HA:RETURN
```

Nel numero **2/88**, nel listato ***Guardian***, a **pagina 66** le linee DATA 16 e 17 sono le seguenti:

```
DATA 88c0203c,00000209,22d851c8,fffc41fa
DATA 01be4a61,203c0000,00f4e200,534032d8
```

Nel listato ***Pangea Basic***, a **pagina 110**, la linea 890 è la seguente:

```
890 DATAC9,85,FF,A9,3F,8D,00,FF,20,5D,FE,A5,FB,85,29,A5,FC,85,28,68,68,EA,A9,
00,8D,00,FF,A2,12,A5,29,A4,28,20,41,16,20,B1,17,48,A9,1F,8D,00,D6,68,8D,01,D6
```

La linea 990 è la seguente:

```
990 DATA00,D6,10,FB,18,20,F2,18,20,70,F9,20,EC,18,4C,24,19,68,68,20,EC,18,4C,
C1,17,00,A6,3D,8E,00,13,A6,3E,8E,09,13,C9,E4,F0,03,4C,67,1A,20,00,03,A9,00,85
```

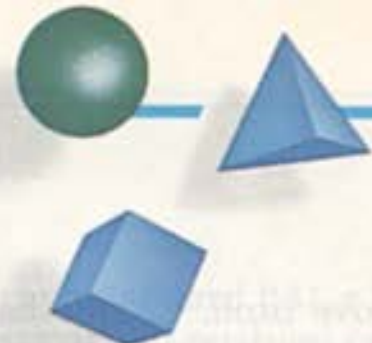
La linea 1000 è la seguente:

```
1000 DATAC1,85,C2,20,0F,88,10,A5,16,69,00,85,C3,A6,30,CA,CA,CA,CA,E4,17,90,33,
CA,E4,17,B0,06,A5,16,C9,FC,B0,20,18,A5,17,69,04,85,C4,20,49,0D,18,A5,02,69,07
```

Come digitare i listati della COMMODORE GAZETTE

Simbolo	Premere
[illegible]	SHIFT CLR/HOME
[illegible]	CLR/HOME
[illegible]	SHIFT CRSR ⇆
[illegible]	CRSR ⇆
[illegible]	SHIFT CRSR ⇅
[illegible]	CRSR ⇅
[illegible]	CTRL 0
[illegible]	CTRL 1
[illegible]	CTRL 2
[illegible]	CTRL 3
[illegible]	CTRL 4
[illegible]	CTRL 5
[illegible]	CTRL 6
[illegible]	CTRL 7
[illegible]	CTRL 8
[illegible]	CTRL 9
[illegible]	F1
[illegible]	SHIFT F1
[illegible]	F3
[illegible]	SHIFT F3
[illegible]	F5
[illegible]	SHIFT F5
[illegible]	F7
[illegible]	SHIFT F7
π	SHIFT ↑
@	*a*
[illegible]	COMMODORE 1
[illegible]	COMMODORE 2
[illegible]	COMMODORE 3
[illegible]	COMMODORE 4
[illegible]	COMMODORE 5
[illegible]	COMMODORE 6
[illegible]	COMMODORE 7
[illegible]	COMMODORE 8

Manoscritti: le collaborazioni dei lettori — manoscritti, disegni e/o fotografie— sono benvenute e verranno valutate in vista di una possibile pubblicazione. Commodore Gazette non si assume comunque responsabilità per perdite o danni al materiale. Si prega di allegare una busta affrancata e indirizzata per ogni articolo. Il pagamento per materiale non richiesto viene effettuato solo in seguito all'accettazione da parte della redazione. I contributi editoriali (di qualunque forma) non si restituiscono. Tutta la corrispondenza editoriale, richieste di annunci, problemi di sottoscrizione abbonamenti, di diffusione e con gli inserzionisti, deve essere indirizzata a: Commodore Gazette - Uffici Editoriali - Via Monte Napoleone, 9 - 20121 Milano
La Commodore Gazette è un periodico indipendente non connesso in alcun modo con la Commodore Business Machines e con tutte le sue sussidiarie e affiliate, compresa la Commodore Italiana S.p.A. La Commodore Gazette viene pubblicata mensilmente dalla IHT Gruppo Editoriale, Via Monte Napoleone 9, 20121 Milano. Il costo degli abbonamenti è il seguente : Italia - 12 numeri L. 84.000, 24 numeri L. 168.000, 36 numeri L. 252.000. Estero - Europa L. 120.000 (10 numeri), Americhe, Asia... L. 180.000 (10 numeri). La redazione si adopera per fornire la massima accuratezza negli articoli e nei listati pubblicati. La Commodore Gazette non si assume responsabilità per eventuali danni dovuti a errori od omissioni.

Classified

Software

Vendo archivio Amiga 1000 (circa 700 dischetti) e computer medesimo causa passaggio a PC 20. Mariangela Gozzi - Via Volta, 68 - Monzambano (MN).

Per l'Amiga: scambio programmi e manuali. Max serietà. Preferibilmente zona Brescia. Scrivere o telefonare a: Alberto Algisi - Via Razziche 5 - 25128 Brescia - Tel. 030/393678.

Cerco amici in zona Pesaro o Siena per scambio programmi, informazioni e materiale vario per l'Amiga 500. Nicola Gentili - Tel. 0577/43444 - 0721/96597.

Superbase italiano originale per l'Amiga vendo, ottimo prezzo. Luca Conti - Piazza Carroccio, 10 - 20025 Legnano (MI) - Tel. 0331/593358 ore 11-15.

Vendo programmi realizzati con il Basic 8.0 per C-128. Gli interessati possono telefonare al numero 081/8954146. Raffaele Di Girolamo - Via Cavour, 4 - 80014 Giugliano (NA).

Cerchi programmi per C-128 (modi 128 e CP/M)? Se sì, telefona allo 011/4150675 chiedendo di Alessandro. Possiedo molti programmi, sia in 80 che in 40 colonne (GEOS 128, Basic 8.0) e giochi (Kickstart, Thai boxing, Pazazz). Non perdere tempo! Annuncio sempre valido.

Amiga scambio eventualmente vendo programmi usati. Es. Terramex, Tetris, Golden Path, Ports of Call e molti altri. Niola Luigi - Via L. Muratori, 2 - 21052 Busto Arsizio (VA) - Tel. 0331/621887.

Amiga software USA Dansoft vende programmi a sole L. 5.000, in abbonamento a L. 4.000. Disk DD/DS a L. 2.500. Disponibile hardware. Danilo D'Agostino - Via Bari, 42 - Villa Raspa (PE) - Tel. 085/4152400.

Svendo causa realizzo centinaia di programmi Amiga, comprese ultimissime novità. Estrema serietà. Il tutto su dischetti DS/DD. Disponibili diversi manuali. Mario Savoia - Via Udine, 22 - 33032 Bertiolo (UD) - Tel. 0432/917520.

Cerco utenti Amiga in provincia di Milano e Como per scambio programmi e informazioni. Danilo Borgonovo - Via Galilei, 3 - Verano Brianza (MI) - Tel. 0362/900057.

Scambio programmi per il C-64/128. Inviare lista a: Vito Lopez - Via Montebello, 21 - 27049 Stradella (PV). Annuncio sempre valido.

Amiga 500/1000/2000 scambiamo software. Disponibili tutte le ultime novità. Arrivi settimanali anche dall'estero. Paperclub - Via E. Kant, 5/F - 20151 Milano - Tel. 02/3013271 (ore pasti o serali). Venditori di software non graditi.

Amiga!!! Sono seriamente interessato allo scambio di software. Si accettano scambi. Scrivere o telefonare a Maurizio Panzani - Via N.Costa,1 - Asti - Tel. 0141/31165.

Scambio software per il C-64/128. Inviare liste a Massimo Zanolo - Via Castello, 3 - 13030 Vintebbio (VC) - Tel. 0163/450617.

Compro giochi ed eventuali istruzioni in italiano per l'Amiga 500. Rispondo a tutti. Spedire le liste a Bruno Zucco - Via Pollino, 16/B - 12040 Fraz. Ronchi (CN).

CLASSIFIED DELLA COMMODORE GAZETTE È UN MODO ECONOMICO PER INFORMARE LA PIÙ VASTA UTENZA COMMODORE SUI VOSTRI PRODOTTI O SERVIZI. GLI ANNUNCI NON A SCOPO DI LUCRO, INVIATI DA PRIVATI, VENGONO PUBBLICATI GRATUITAMENTE (COMPILARE L'APPOSITA SCHEDA DI SERVIZIO LETTORI).

Quote: 15.000 lire per linea, minimo 4 linee. Aggiungere 5.000 lire per ogni parola in grassetto o 50.000 lire per l'intero annuncio in grassetto.

Condizioni: pagamento anticipato. Vengono accettati assegni e vaglia postali. Gli assegni devono essere intestati a: IHT GRUPPO EDITORIALE s.r.l.

Forma: gli annunci sono soggetti all'approvazione dell'editore e devono essere scritti a macchina o in modo molto chiaro. Una linea equivale a 40 lettere, spazi tra le parole compresi. Pregasi sottolineare le parole che si intendono scrivere in grassetto.

Informazioni generali: gli inserzionisti devono sempre specificare nome e indirizzo completo. Gli annunci appariranno nel primo numero disponibile dopo il ricevimento.

Data di chiusura: il 10 del mese precedente alla data di copertina (per esempio il numero di iugno chiude il 10 maggio).

Inviare il materiale a:

IHT GRUPPO EDITORIALE
UFFICI PUBBLICITARI
VIA MONTE NAPOLEONE 9
20121 MILANO

Attenzione: La Commodore Gazette non si assume responsabilità in caso di reclami da parte degli inserzionisti e/o dei lettori. Nessuna responsabilità è altresì accettata per errori e/o omissioni di qualsiasi tipo.

Scambio programmi per il C-64 solo su disco. Massima serietà. Rispondo a tutti. Annuncio sempre valido. Palmiero Lattanzio - Via Machiavelli, 26 - 66054 Vasto (CH) - Tel. 0873/59963.

Scambio programmi per l'Amiga. Posseggo vasto archivio con continui aggiornamenti. Massima serietà. Renzo Vincenzi - Via Provinciale, 33 - 28050 Arizzano (NO) - Tel. 0323/551254.

Amiga vendo programmi originali per cessione hobby, a metà prezzo. Titoli: Page Setter italiano, Deluxe Paint 780 x 560, Express Paint 2.0, Bioritmo, Danlotto, Memory Alpha disk, Velox soft, Amiga textloper, ecc. Danilo D'Agostino - Tel. 085/4152400.

Incredibile! vendo 500 dischi pieni zeppi di software per il C-64 a L. 500.000. Solo in blocco e non trattabili. Telefonare dopo le ore 20.00 a Oreste Natale - Via Quadrivio, 38 - 84020 Campagna (SA).

Amiga! Scambio programmi. Max serietà. Sempre valido. Inviare liste a Luigi Fortino - Via Torino, 3 - 84092 Bellizzi (SA).

Cerco programmi di utilità per il C-128, sia gestionali che grafici, solo in modo 128 e CP/M. Inviare liste e richieste. Patrizia Burani - Via Budapest, 7 - 43100 Parma.

Per Amiga, C-64 e PC, cambio i migliori programmi. Inviare la lista completa su richiesta (inviare francobolli per risposta). Scrivere o telefonare a Gianluca Angelici - Via Capestro, 10 - 63023 Fermo (AP) - Tel. 0734/622759.

Scambio programmi per l'Amiga, cerco manuali possibilmente in italiano, scrivere a Carlo Cazzavillan - Via Vicenza, 39 - 36071 Arzignano (VI).

La più vasta libreria software in Italia con agg. settimanale!!! Iscrizione gratuita con bollettini mensili! L. 5.500 a programma. Abbonamenti. Hardware. - Amiga Club *Europeo* - C.so Trento, 63 - 38086 Pinzolo (TN) - Tel. 0465/51610!!!

Hardware

Vendo C-128D + MPS 803 + 100 dischi + riviste, libri, joystick, carta per stampante e tanto altro a L. 950.000. Ottimo stato e imballi originali. Pomeriggio o sera. Adriano - Tel. 06/6070341.

Vendo stampante Commodore MPS 803. Un anno di vita. Usata pochissimo. Completa di trattori. Regalo di un

nuovo. L. 350.000 trattabili. Telefonare in ore pasti al 0862/66468. Chiedere di Maurizio.

Attenzione!!! È disponibile l'emulatore hardware-software del C-64 per l'Amiga in versione originale della ReadySoft americana a L. 40.000!!! Ora anche in versione per Amiga 1000!!! Affrettatevi!!! Scrivere o telefonare a Marco Costantino - Corso Trento, 63 - 38086 Pinzolo (TN) - Tel. 0465/51610.

Vendo stampante Epson EX800, 300 caratteri al secondo, due modi NLQ, 80 colonne. Funziona con driver JX-80 sull'Amiga per produrre grafica ottima opzionalmente a colori. Come nuova. Visionabile e provabile. Luigi Callegari - Via De Gasperi, 47 - 21040 Sumirago (VA) - Tel. 0331/909183 (sera).

A2000 Commodore Ital. 2 mesi, 2 drive, mon. 1084, possibile exp 512K, hard nofastmem, L. 2.800.000 trattabili. Raffaello - Tel. 049/5992056 (sera).

Volete un bel computer? Regalatevi un C-128 superaccessoriato! Ho MPS-803, drive 1570, registratore, decine di nastri, 200 disk con prg per 64-128-CP/M, digitalizzatore voce umana, cartucce... Assicuro max disponibilità per prove e spiegazioni! Luca Lorenzini - Via Lumumba, 11 - 41011 Campogalliano (MO) - Tel. 059/525861.

Come nuovo vendo C-64 + drive 1541 + MPS 803 + mouse + ad. telematico + riviste + accessori + 1.500 prg su 200 disk e 150 cassette + corso di grafica, a L. 2.000.000 trattabili. Vendo con manuali e imballaggi originali, pagato L. 3.500.000.

Cendo stampante Panasonic KX-P1091 non danneggiata in buone condizioni. Ottima per IBM e altri personal come il C-64/128, l'Amiga... Buon prezzo trattabile. Inoltre vendo/scambio ottimo software. Contattatemi. Giovanni Alongi - Via Edison, 81 - 92023 C/Bello (AG).

Vendo Vic-20 solo tastiera e modulatore TV + 20 cassette di giochi, perfettissimo, a L. 100.000 trattibili. Giampiero Cozzolino - Via Libertà, 209 - 80055 Portici (NA).

Vendo modem-phone 300 baud di potenza con cavetto Amiga, o scambio con manuale Amiga macroassembler della Metacomco. Telefonare 02/2154095 Luca.

Vendo C-64, drive 1541, registratore e programmi vari a L. 300.000 circa. Gian Luca Nanuzzi - Via Franchetti, 7 - 40141 Bologna - Tel. 473464.

Amiga 1000 vendo con eventuale monitor e stampante. Per informazioni telefonare allo 075/5001200. Federico Luccioli.

Vendo Commodore 128, giugno '87, registratore, joystick + 20 cassette. Affarone, solo 499.000 trattabili! Tutto come nuovo! Mario Savoia - Via Udine, 22 - 33032 Bertiolo (UD) - Tel. 0432/917520.

Vendo C-64 + C2N + 1541 + MPS 801 + cartuccia velocizzatrice + circa 200 dischi + 100 cassette + manuali vari per L. 850.000 trattabili. Solo zona Bologna. Roberto Michelini - Via Ronzani, 61 - 40033 Casalecchio di Reno (BO) - Tel. 051/578646 (ore 18.30 - 20.30).

Vendo C-128D + monitor 1901 + modem 64 + mouse + registratore C2N + cartuccia partner 128 + corso completo "Video Basic" + 40 dischi zeppi di giochi e utility, il tutto con manuali originali a L. 1.500.000. Regalo inoltre più di 50 cassette con videogames. Telefonare allo 0823/913145 e chiedere di Luigi. Max serietà.

Cerco Vic-20 anche in cattivo stato, purché funzionante e di prezzo non superiore alle 90.000. Telefonare allo 0377/84334 ora di cena.

Vendo C-128D + stampante MPS 803 + monitor Philips a fosfori verdi + joystick + copritastiera + espansione grafica 640 x 200 + Basic 8.0 + manuali + software. Il tutto a L. 500.000. Telefonare allo 0377/84334 ora di cena.

Vendo per C-64/128 - per cambio sistema - programmatore di EPROM PP64 L.150.000, final cartridge III L. 70.000 e Niki II L. 60.000. Telefonare dopo le ore 20.00 allo 049/703394.

Disponibile l'emulatore 64 (hardware/software) in versione originale, della Ready Soft a L. 40.000. Affrettatevi. Scrivete o telefonate a Riccardo Torazza - Via Canale, 29 - Venaria (TO) - Tel. 011/495417.

Varie

A tutti gli utenti del C-128. A pag. 80044000 del Videotel, trovate la rubrica "A modo mio" e un software club dove tutto è rigorosamente gratuito. Per informazioni scrivete a Libero Zanetti - Via Fornasini, 26 - 40128 Bologna.

Vorrei corrispondere con utenti C-128 per nuove amicizie di lettere e computer da tutt'Italia. Vi aspetto. Intanto accendo la stampante Bit. Scrivere a: for Alex c/o Alfonso Roselli - Via Sannio, 32 - M.Silvano (PE).

Sarei grato se qualche amico mi insegnasse a superare il 50% di Barbarian (per intenderci, quando c'è quel mago che lancia palle di fuoco). Francesco Mai - Via Campagnola, 2 - 37011 Bardolino (VE).

Vendo i seguenti libri per il C-64: Come usare il tuo C-64 Vol.I e Vol.II. Corso di Assembler II con cassetta. Corso di grafica. Computer hardware (Vic-20, C-64, Sinclair). Telefonare allo 081/923810 (Antonio).

Hai un modem? Incontriamoci allora su SKYNET attiva ogni giorno dalle 21.00 alle 7.00. Parametri: 300 baud 8/N/I. Telefono 0836/569195. SKYNET ti offre tra l'altro un'area hackers con informazioni esplosive. Che aspetti allora a chiamare?

Offresi lavoro di imbustamento/spedizione e archivio al proprio domicilio. Chiunque fosse interessato può richiedere informazioni senza impegno a Fabrizio Svieri - Via Pioppelle, 5 - 44036 Francolino (FE). Si prega di allegare bollo per risposta, grazie.

Cerco possessori del campionatore EMAX per formare gruppo d'acquisto e scambio suoni, inoltre cerco software musicale per Atari 1040. Se interessati telefonate o scrivete a Ugo Bolzoni - Via S.Bellino, 8 - 45100 Rovigo - Tel. 0425/22619.

Circuiti integrati originali per tutta la serie Commodore, Kit di montaggio elettronici. Piccolo hardware. Scheda diagnostica per C-64 e 1541 per individuare guasti. **Catalogo gratis. Tel. 055/608440**.

Commodore Club

Arcadia Amiga Group cerca nuovi amici in Udine e provincia. Vasta biblioteca software. Siamo interessati a tutto quanto riguarda A500/A2000 (massima serietà). Tel 0432/660623-541240.

Telefonate all'Halley Amiga Club di Recanati, unico nella zona, per scambio di software, di idee. Luigi 071/980180; Michele 071/9841262.

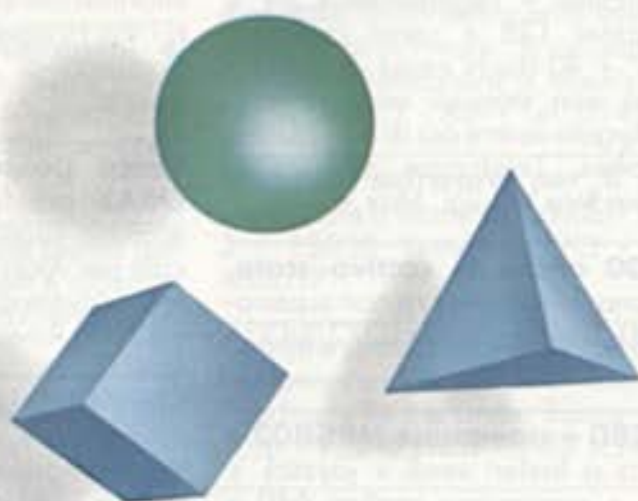
Cielo Commodore Club in attività dal 1983, cerca aderenti e collaboratori per incrementare le sue iniziative. Disponibile un'enorme quantità di programmi per Vic-20, C-64, C-128 & Amiga più un cospicuo numero di libri e riviste per consultazioni.

Si sono aperte le iscrizioni per il nuovo club CBM 64/128/A500/A2000. Vastissimo software, massima serietà,. Telefonare allo 0578/294012.

Amiga club Cremona cerca appassionati fantasiosi di omonimo personal per creazione di ottimo rapporto di scambio e collaborazione. Fantastiche novità! Roberto Romano - Via Gradisca, 5 - 26100 Cremona - Tel. 0372/24929.

Commodore Club Terontola per i computer CBM C-64/128/A500, ha aperto le iscrizioni. A disposizione più di 2000 programmi per C-64/128 e 1000 programmi per l'A500. Per l'iscrizione telefonare allo 0575/67616. Massima serietà.

INDICE DEGLI INSERZIONISTI

Inserzionista	Pag.
Bytec	37
Commodore Italiana	50, 51, 52
Desme Universal	88
G.P. System	115
IHT Gruppo Editoriale	1, III, IV
Lago	6
Pix Computer Service	93
Sim Hi Fi	II

Direzione vendite spazi pubblicitari:

IHT
Gruppo Editoriale S.r.l.
Commodore Gazette
Uffici Pubblicitari
Via Monte Napoleone, 9
20121 Milano
Tel. 02/794181 - 799492
Telex 334261 IHT I
Telefax 784021

Questo indice è da considerarsi come un servizio addizionale. L'Editore non si assume alcuna responsabilità per eventuali errori e/o omissioni.

Indirizzare eventuali lamentele riguardanti gli inserzionisti a:

Commodore Gazette
Uffici Pubblicitari
Relazioni Inserzionisti
Via Monte Napoleone, 9
20121 Milano

Nessuna responsabilità viene altresì assunta dalla Commodore Gazette per eventuali problemi di qualsiasi natura con gli inserzionisti. La responsabilità di quanto pubblicato negli spazi pubblicitari è esclusivamente del committente.

NEL PROSSIMO NUMERO

Luglio/Agosto 1988 (in edicola dal 31 agosto)

- **Prove hardware: le stampanti -** Le più diffuse stampanti messe alla prova di fronte alle principali esigenze dell'utente. Velocità e flessibilità d'impiego, costi, ingombro, qualità della stampa e compatibilità. Un istruttivo confronto hardware.

- **Corso di programmazione -** Penultima puntata del nostro corso sul linguaggio macchina, che costituisce ormai un vero e proprio libro sulla programmazione a basso livello del C-64.

- **Comete -** Ancora un tema astronomico per il C-128: il calcolo delle coordinate di comete e asteroidi, con la possibilità di catalogare fino a 200 corpi celesti.

- **Mondo Amiga -** Continuano le nostre esplorazioni negli Stati Uniti, alla ricerca di programmatori e tecnici, esperti ed editori, che possano tenerci aggiornati in tempo reale sulle novità del mondo Amiga.

- **GEOS, un sistema in multitasking -** Terza parte della serie dedicata a GEOS: questa volta affronteremo le strutture di icone e menu, dando una prima dimostrazione concreta delle funzioni svolte da MainLoop.

SERVIZIO LETTORI

Questa scheda è valida fino al 31 agosto 1988

A. Come giudica questo numero di Commodore Gazette?

- ☐ 1. Ottimo
- ☐ 2. Molto buono
- ☐ 3. Buono
- ☐ 4. Discreto
- ☐ 5. Sufficiente
- ☐ 4. Mediocre
- ☐ 6. Insufficiente

B. Quale(i) articolo(i) di questo numero ha apprezzato maggiormente?___

C. Quale(i) articolo(i) di questo numero giudica peggiore(i)?___

D. Quali argomenti dovrebbero essere trattati nei prossimi numeri di Commodore Gazette?___

E. Con quale aggettivo descriverebbe Commodore Gazette?___

F. Quante persone leggono la sua copia di Commodore Gazette?

- ☐ 1. Uno
- ☐ 2. Due
- ☐ 3. Tre
- ☐ 4. Quattro o più

G. Ha dei suggerimenti?___

H. Quale(i) computer utilizza?

- ☐ 1. C-64
- ☐ 2. C-128
- ☐ 3. C-128D
- ☐ 4. Amiga 500
- ☐ 5. Amiga 1000
- ☐ 6. Amiga 2000
- ☐ 7. PC 1
- ☐ 8. PC 10
- ☐ 9. PC 20
- ☐ 10. Altro (specificare)___

I. Quale(i) computer intende acquistare nel futuro?

- ☐ 1. C-64
- ☐ 2. C-128D
- ☐ 3. Amiga 500
- ☐ 4. Amiga 2000
- ☐ 5. PC 1
- ☐ 6. PC 10
- ☐ 7. PC 20
- ☐ 8. Altro (specificare)___

L. È un acquirente dei libri della IHT? Se sì come li giudica?___

M. Quali altre riviste (sia d'informatica che non) legge abitualmente?___

N. Indichi quali sono i suoi maggiori interessi

- ☐ 1. Videoregistrazione
- ☐ 2. Hi-Fi
- ☐ 3. Fotografia
- ☐ 4. Automobili
- ☐ 5. Sport
- ☐ 6. Viaggi

Nome ___
Cognome ___
Indirizzo ___
Città ___
Prov. ___ **C.a.p.** ___
Età ___ **Professione** ___

COMMODORE GAZETTE **giugno 1988**

SCHEDA ORDINAZIONE LIBRI

Con il presente tagliando desidero ordinare i seguenti libri:

- ☐ L'Amiga (Michael Boom) .... L. 60.000
- ☐ Il Manuale dell'AmigaDOS (Commodore-Amiga) .... L. 60.000
- ☐ Programmare l'Amiga Vol. I (Eugene P. Mortimore) .... uscita: aprile '89
- ☐ Programmare l'Amiga Vol. II (Eugene P. Mortimore) .... uscita: ottobre '88
- ☐ Guida ufficiale alla programmazione di GEOS (Berkeley Softworks) .... L. 64.000
- ☐ Flight Simulator Co-pilot (Charles Gulick) .... L. 30.000
- ☐ Volare con Flight Simulator (Charles Gulick) .... L. 45.000

Pagherò in contrassegno al postino la somma di L. .... + spese postali (L. 5.000 per volume)

Nome ___
Cognome ___
Indirizzo ___
Città ___
Prov. ___ **C.a.p.** ___ **Tel.** ___

☐ **Desidero inserire gratuitamente un mio annuncio nella rubrica CLASSIFIED (solo per i privati e per gli annunci non a scopo di lucro).**

☐ **Desidero inserire un mio annuncio nella rubrica CLASSIFIED. Allego assegno bancario o circolare o postale oppure fotocopia della ricevuta del vaglia postale per un totale di L. ____________________ . Il mio codice fiscale o partita IVA (per le aziende) è il seguente: ____________________.**

Attenzione: perché un annuncio venga accettato è necessario che sia stato compilato anche il questionario presente sull'altro lato di questo tagliando.

TESTO: ____________________

Inserire all'interno di una busta affrancata e spedire a:

Commodore Gazette
Servizio Lettori
Via Monte Napoleone, 9
20121 Milano

Inserire all'interno di una busta affrancata e spedire a:

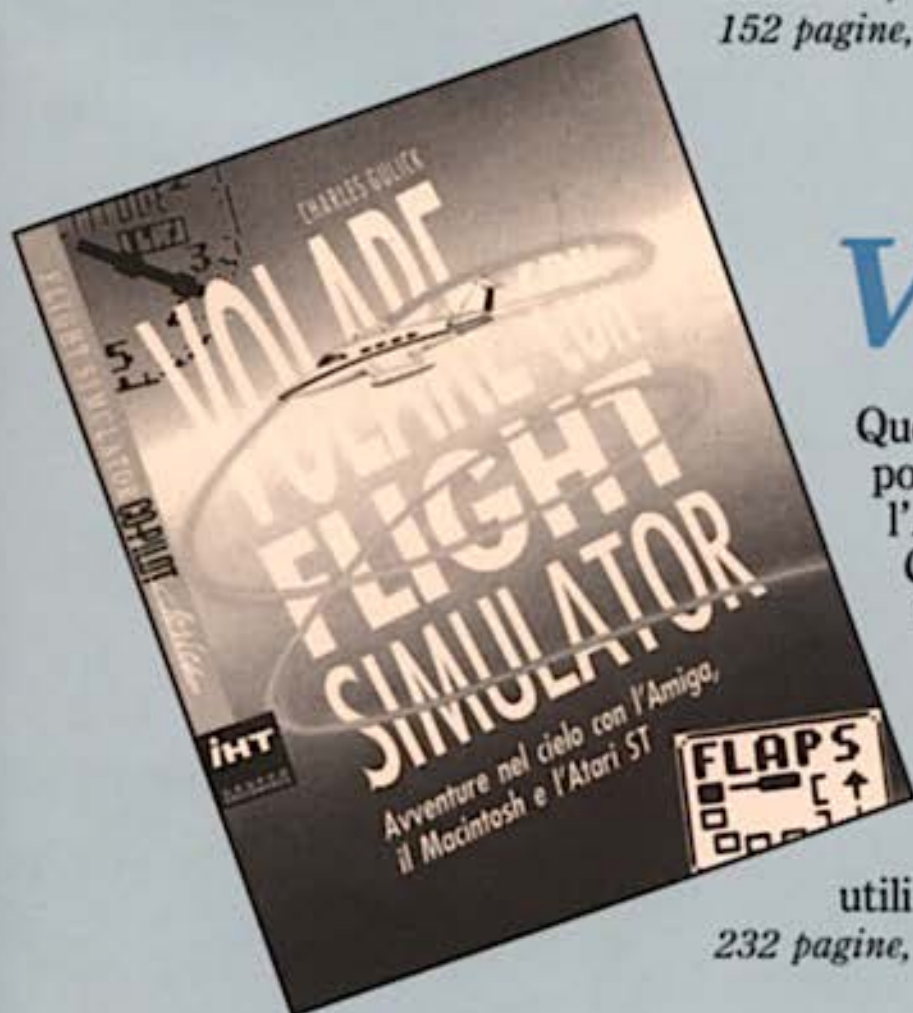
IHT Gruppo Editoriale
Divisione Libri
Via Monte Napoleone, 9
20121 Milano

UN COMPUTER, UN LIBRO...
...LE ALI

FLIGHT SIMULATOR CO-PILOT

Questo volume è un vero "istruttore di volo" per coloro che possiedono il programma Flight Simulator della Microsoft per i PC IBM e compatibili, oppure Flight Simulator II della SubLogic per Commodore 64, Apple II, Atari 800 XL e XE.

Non è un semplice compendio di comandi da ricordare a memoria, ma un brillante compagno d'avventura, scritto con estro e ironia. Un libro da sfogliare con il computer acceso, per imparare passo dopo passo i principi del volo "rettilineo e livellato", le procedure di decollo e di atterraggio, del volo strumentale e notturno, e del volo acrobatico.

152 pagine, L. 30.000

VOLARE CON FLIGHT SIMULATOR

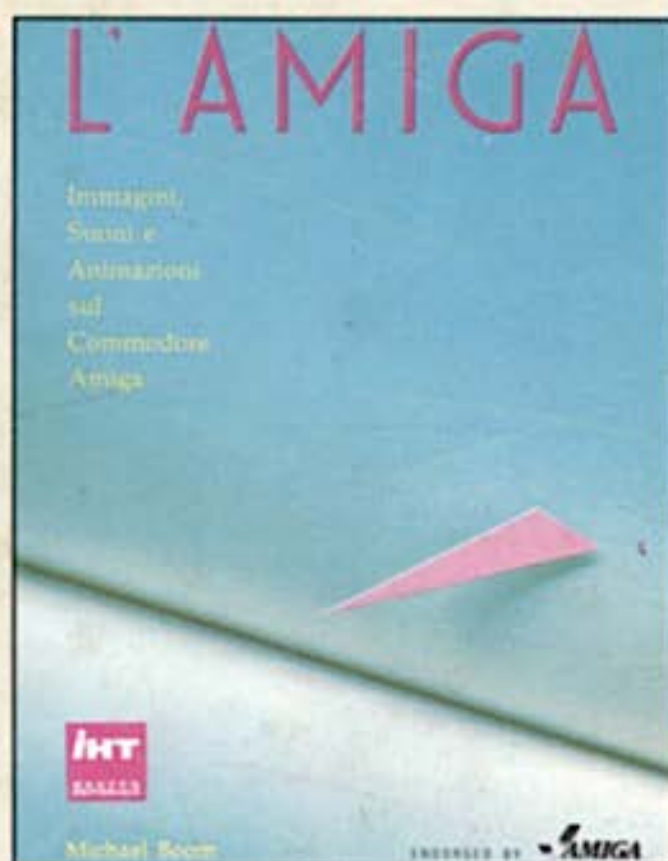
Questo volume si rivolge a tutti gli utenti che possiedono la versione di Flight Simulator per l'Amiga, l'Atari ST o l'Apple Macintosh.

Con uno stile spigliato, ma con grande attenzione a ogni dettaglio, l'autore conduce l'allievo pilota attraverso le varie fasi di un corso di volo che si concluderà con un primo approccio al volo acrobatico. Il testo è completato da dettagliate cartine di rotta, schemi che illustrano l'uso dei comandi, parametri di volo ideali e procedure utili nelle più svariate situazioni di volo.

232 pagine, L. 45.000

I libri IHT sono disponibili nelle migliori librerie e computer shop. Per ordini diretti servirsi dell'apposito modulo pubblicato a pagina 127.

IHT Gruppo Editoriale - Via Monte Napoleone, 9 - 20121 Milano - Tel. 02/794181-794122 - Fax 784021 - Telex 334261 IHT I
Distribuzione: Messaggerie Libri - Via Giulio Carcano, 32 - 20141 Milano - Tel. 02/8438141

Un computer chiamato AMIGA

L'AMIGA

Come il Commodore Amiga ha cambiato il mondo dei computer, così il volume *L'Amiga* proietterà la vostra immaginazione lungo nuovi orizzonti di creatività. Preparatevi a ottenere stupefacenti risultati dal vostro computer: sofisticate immagini video, suoni e musica, sequenze animate da registrare su videocassetta, e molto di più. In questo volume troverete inoltre consigli utili per ottenere il meglio dall'Amiga Basic e da prodotti software eccezionali come Deluxe Paint, Deluxe Music e Deluxe Video.

416 pagine, oltre 100 illustrazioni, L. 60.000

IL MANUALE DELL'AMIGADOS

Questo volume è l'unica documentazione ufficiale realizzata dalla Commodore sul sistema operativo dei computer Amiga. Contiene tre libri (Il Manuale per l'utente, Il Manuale per il programmatore e Il Manuale di riferimento tecnico) che costituiscono nel loro complesso la guida più completa per ogni utente dell'Amiga, dal principiante che vuole usare i comandi dell'AmigaDOS (che non sono spiegati dall'opuscolo in dotazione al computer), fino al programmatore evoluto che troverà utili informazioni per programmare in C e in Assembly.

376 pagine, L. 60.000

I libri IHT sono disponibili nelle migliori librerie e computer shop. Per ordini diretti servirsi dell'apposito modulo pubblicato a pagina

IHT Gruppo Editoriale - Via Monte Napoleone, 9 - 20121 Milano - Tel. 02/794181-794122 - Fax 784021 - Telex 334261 IH

Distribuzione: Messaggerie Libri - Via Giulio Carcano, 32 - 20141 Milano - Tel. 02/8438141